



Departamento de Ciências e Tecnologias de Informação

Sistema Web e Mobile para Apoio a Gestão de Projetos de Sistemas de Informação

Carmem Fernandes Tavares

Dissertação submetida como requisito parcial para a obtenção de grau de Mestre em
Engenharia Informática

Orientador:

Prof. Doutor Carlos Serrão

Professor Auxiliar, ISCTE-IUL

Outubro 2013

Resumo

A forma como um projeto é gerido tem vindo a evoluir ao longo dos tempos. Com a crescente preocupação em entregar os projetos dentro dos prazos estipulados, com qualidade e dentro do orçamento previsto, foi necessário melhorar a forma de gerir os mesmos. Neste sentido têm vindo a surgir várias metodologias e, principalmente, ferramentas informáticas que apoiam os gestores de projetos a gerir os mesmos. Tendo em conta que gerir projetos de desenvolvimento de sistemas de informação engloba desafios específicos dos projetos desta natureza e, no sentido de ajudar os gestores a enfrentarem tais desafios com sucesso, propôs-se efetuar o desenvolvimento de uma aplicação web e mobile de apoio a gestão de projetos desta natureza.

Este trabalho procura fazer um mapeamento entre as principais metodologias de gestão de projetos, principalmente, as direcionadas para projetos de desenvolvimento de sistemas de informação, e os grupos de processos e áreas de conhecimento de gestão de projetos definidos no PMBOK. Toda essa informação é disponibilizada numa aplicação web, que está, igualmente, acessível nos dispositivos móveis.

Com esta aplicação web e mobile, pretendeu-se desenvolver uma aplicação informática interativa de apoio a gestão de projetos de sistemas de informação, tendo em conta a metodologia adotada para a gestão do projeto e as linhas de orientação do PMBOK.

Palavras – Chaves: Gestão de Projetos, PMBOK, Metodologias de Gestão de Projetos, Metodologias de Desenvolvimento de Software, Ferramentas de Gestão de Projetos, Aplicações Móveis.

Abstract

The way a project is managed has evolved over the time. With the growing concern in delivering the projects within the stipulated deadlines, with quality and within budget, it has become necessary to improve the way projects are managed. In this sense had emerged several methodologies and software tools that primarily support the project managers to manage them. Given that managing information systems development projects brings with it specific challenges, and in order to help managers cope with them successfully this work proposes the development of a web and mobile application to support the management of projects of this nature.

This work seeks to make a mapping between the main methodologies of project management, especially targeted for the development of information systems projects and the process groups and knowledge areas of project management defined in the PMBOK. All this information is available in a web application that is also accessible on mobile devices.

With this mobile and web application, it was developed an interactive software application supporting project management information systems, taking into account the methodology adopted for the project management and the guidelines of the PMBOK.

Keywords: Project Management, PMBOK, Project Management Methodologies, Software Development Methodologies, Tools Project Management, Mobile Application.

Agradecimentos

A Deus por me ter dado saúde e força de vontade o suficiente para levar este projeto até ao fim.

Ao meu orientador, Prof. Doutor Carlos Serrão, pelo profissionalismo, motivação, disponibilidade e todo o apoio prestado, mesmo quando aparecia no seu gabinete sem aviso prévio.

Aos meus pais Francisco e Alcinda pela minha existência, e por tudo o que fizeram para que eu seja aquilo que eles nunca tiveram oportunidade de ser.

Aos meus irmãos Maria, António, Silvina, José, Cecília, Linda, Ermelinda e Aquilino pela amizade, companheirismo e sobretudo pela união e laço familiar e um especial obrigado ao meu irmão António e à minha irmã Linda por terem sido os responsáveis pela minha passagem pelo ensino superior.

À minha sobrinha Sheila pela preciosa ajuda na tradução de artigos.

Ao meu amigo Zé Mário pela amizade, pela ajuda na tradução e todo o apoio dado.

À minha coordenadora Dr. Isabel Baía pelo apoio que me prestou sempre que solicitado.

Às minhas amigas e colegas de trabalho pela motivação.

À Sra. Fátima Silva do DCTI pela simpatia na prestação de informações e ajuda na resolução de certas burocracias ao longo do mestrado.

A todos os que de uma forma ou de outra contribuíram para a realização deste projeto.

Índice

Resumo	III
Abstract	IV
Agradecimentos	V
Lista de Figuras	VIII
Lista de Tabelas	IX
Abreviaturas	X
1 Introdução	1
1.1 Motivação.....	1
1.2 Questões de Investigação	2
1.3 Objetivo	3
1.4 Métodos de Investigação	4
1.5 Organização do Documento.....	4
2 Estado da Arte	6
2.1 Introdução à Gestão de Projetos	6
2.1.1 PMBOK Guide	6
2.1.2 Definição do Projeto.....	7
2.1.3 Gestão de Projetos (GP)	7
2.1.4 Ciclo de Vida do Projeto	11
2.2 Metodologias de Gestão de Projetos de SI	12
2.2.1 Rational Unified Process (RUP)	14
2.2.2 PRINCE2 (Projects in Controlled Environments – 2ª versão)	15
2.2.3 TenStep.....	18
2.2.4 Extreme Programming (XP).....	20
2.2.5 SCRUM.....	21
2.3 Ferramentas de Gestão de Projetos (FGP)	23
2.3.1 Análise das Ferramentas Desktop	23
2.3.2 Análise das Ferramentas Baseadas na Web (<i>Web-Based</i>)	25
2.3.2.1 Análise das Ferramentas de Suporte ao Desenvolvimento Ágil	28
2.3.4 Estudo Comparativo	30
2.3.4.1 Comparação entre as Ferramentas que Suportam os Processos de Desenvolvimento Ágil.....	33

2.4 Conclusão	35
3 Mapeamento do PMBOK com as Metodologias de GP	36
3.1 Áreas de Conhecimento e Processos do PMBOK.....	36
3.2 Gestão de Projetos de SI	38
3.3 Mapeamento entre o RUP e o PMBOK	43
3.4 Mapeamento entre o PRINCE2 e o PMBOK	45
3.5 Mapeamento entre o Extreme Programming (XP) e o PMBOK	47
3.6 Mapeamento entre o SCRUM e o PMBOK	49
3.7 Conclusão	52
4 Processo de Implementação da Aplicação.....	54
4.1 Arquitetura do Sistema	54
4.2 Requisitos Funcionais da Aplicação.....	59
4.3 Protótipo da Aplicação	60
5 Validação da Aplicação.....	68
5.1 Atores e Casos de Uso	68
5.2 Validação dos Requisitos.....	75
6 Conclusão e Trabalho Futuro	77
Referências	80
Anexos	83
Anexo A – Detalhes dos Processos do PMBOK Guide	83
Anexo B – Mapeamento detalhado entre o RUP e o PMBOK.....	85
Anexo C – Mapeamento detalhado entre o SCRUM e o PMBOK.....	87

Lista de Figuras

Figura 1: Grupos de processos da Gestão de Projetos. Fonte: (PMBOK).....	9
Figura 2: Áreas de conhecimento da Gestão de Projetos	9
Figura 3: Fases do ciclo de vida de um projeto. Fonte: (PMBOK).....	11
Figura 4: RUP. Fonte: (IBM, 2012)	14
Figura 5: Fluxo de iterações do RUP. Fonte: (IBM, 2012)	15
Figura 6: Processos do PRINCE2. Fonte: (PRINCE2, 2012b).....	16
Figura 7: Passos da metodologia TenStep. Fonte: (TenStep, 2012b).....	20
Figura 8: Processo Scrum. Fonte: (WKP)	22
Figura 9: Processos de Gestão de Projetos. Fonte: (adaptado de PMBOK).....	36
Figura 10: Resumo gestão de custos do projeto. Fonte: (PMBOK)	37
Figura 11: Interação entre grupo de processos e fases ou projeto. Fonte: (PMBOK)....	38
Figura 12: Ilustração das possíveis falhas no processo de desenvolvimento de software.	40
Figura 13: “Write Once, Run Everywhere”. Fonte: (Sencha Touch, 2013).....	55
Figura 14: Arquitetura MVC do Sencha Touch. Fonte: (Sencha Touch, 2013).....	56
Figura 15: Estrutura do projeto em Sencha Touch	56
Figura 16: Arquitetura da aplicação desenvolvida	57
Figura 17: Ecrã inicial do utilizador	61
Figura 18: Ecrã inicial de administrador	62
Figura 19: Menu de opções das metodologias.....	63
Figura 20: Outras opções (menu) da aplicação	64
Figura 21: Visualização das metodologias na página inicial no desktop	65
Figura 22: Atores e Casos de Uso	68
Figura 23: Ecrã de registo e autenticação	69
Figura 24: Visualização do perfil de utilizador	70
Figura 25: Ecrã inicial da aplicação	71
Figura 26: Visualização do mapeamento do PPRINCE2 com grupo de processos e áreas de conhecimento	72
Figura 27: Visualização das áreas de conhecimento e grupo de processos.....	73
Figura 28: Atualização dos dados no sistema.....	74
Figura 29: Visualização da lista de utilizadores e os seus dados.....	75

Lista de Tabelas

Tabela 1: Mapeamento de grupos de processos e áreas de conhecimento da Gestão de Projetos. Fonte: (Adaptado do PMBOK)	10
Tabela 2: Comparison of various methodologies from a project management perspective.	13
Tabela 3: Comparação das ferramentas analisadas	32
Tabela 4: Comparação ao nível de gestão de requisitos (baseado em (Pinto, 2010))	34
Tabela 5: Comparação ao nível de gestão de qualidade (baseado em (Pinto, 2010))	34
Tabela 6: Comparação ao nível de gestão de tempo (baseado em (Pinto, 2010)).....	34
Tabela 7: Comparação ao nível de gestão de pessoas (baseado em (Pinto, 2010)).....	35
Tabela 8: Mapeamento entre Áreas de Conhecimento e Metodologias	42
Tabela 9: Mapeamento entre Grupo de Processos e Metodologias.....	43
Tabela 10: Mapeamento entre o RUP e o PMBOK	44
Tabela 11: Mapeamento entre o PRINCE2 e o PMBOK	47
Tabela 12: Mapeamento entre o XP e o PMBOK	48
Tabela 13: Mapeamento entre o SCRUM e o PMBOK	51
Tabela 14: Requisitos funcionais da aplicação.....	60
Tabela 15: Validação dos requisitos.....	75

Abreviaturas

AJAX - Asynchronous JavaScript + XML

API - Application Programming Interface

BD – Bases de Dados

CIO – Chief Information Officer

CCTA - Central Computer and Telecommunications Agency

CSS - Cascading Style Sheets

CPM – Critical Path Method

EAP – Estrutura Analítica de Projeto

GP - Gestão de Projeto

GPL – General Public License

HTML - Hypertext Markup Language

iOS - Sistema operativo móvel (iphone OS)

JSON - JavaScript Object Notation

LDAP - Lightweight Directory Access Protocol

MVC – Model View Controller

PERT – Program Evaluation and Review Technique

PMBOK - Project Management Body of Knowledge

PMI - Project Institute Management

RUP - Rational Unified Process

SDK - Software Development Kit

SI - Sistemas de Informação

SWOT – Strengths Weaknesses Opportunities Threats

TI – Tecnologias de Informação

UI - User Interface

UML - Unified Modeling Language

WBS - Work Breakdown Structure

WebDAV - Web Distributed Authoring and Versioning

WKP - Wikipedia

WWW - World Wide Web

XP - Extreme Programming

1 Introdução

1.1 Motivação

No exercício de gestão de projetos existe um conjunto muito importante de variáveis que devem ser consideradas, assim como um conjunto de intervenientes e participantes que devem ser geridos, para que o projeto seja bem-sucedido.

Todos estes elementos devem funcionar de uma forma integrada, e devem poder comunicar entre si de forma clara e simplificada, para levar o projeto a bom termo.

Assim, uma das principais motivações para a realização deste trabalho passa por tornar acessível aos utilizadores, através de um *browser* web no seu desktop ou dispositivo móvel, a informação relativa à relação entre os processos e áreas de conhecimento do PMBOK e metodologias, permitindo um serviço de apoio cómodo e prático no processo de gestão de projetos de desenvolvimento de SI, tendo em conta a metodologia de gestão de projeto adotada.

Outra das motivações prende-se com a evolução e o impacto das tecnologias móveis a nível mundial e nacional. Diz um estudo da Accenture (Accenture, 2012):

“O impacto das estratégias de mobilidade nas empresas excederá o da emergência da Internet, durante os anos 90”, (Accenture, 2012)

As tecnologias móveis, os *smartphones*, e os *tablets* estão a revolucionar a forma como as empresas desenvolvem os seus negócios. De acordo com dados publicados (em agosto de 2012) pela IDC¹, em 2011 as vendas mundiais de *smartphones* e de *tablets* ultrapassaram as vendas de computadores pessoais, e em 2012, este intervalo deveria alargar-se. As vendas de equipamentos móveis (*smartphones* e *tablets*) vão ascender a quase 800 milhões de unidades (686 milhões de *smartphones* e 107 milhões de *tablets*), enquanto as vendas de computadores não deverão exceder os 400 milhões (Ionline, 2012). Este fenómeno deve-se também ao crescimento exponencial das aplicações móveis.

¹ A IDC é a empresa líder mundial na área de "market intelligence", serviços de consultoria e organização de eventos para os mercados das Tecnologias de Informação, Telecomunicações e Eletrónica de Consumo. <http://www.idc.pt>

Apesar destes equipamentos móveis ainda serem em grande parte adquiridos e “consumidos” de forma pessoal, foi após o lançamento do iPhone 3GS em 2008 que as organizações começaram a compreender o que seria possível com aplicações móveis corporativas, quer para processos internos, quer para contacto com clientes e parceiros.

A dimensão do equipamento, a popularidade da interface com o utilizador, o *browser* web e a emergência do mercado de aplicações foram aspetos cruciais que permitiram que as organizações comesçassem a explorar as oportunidades que as aplicações móveis podem proporcionar, nomeadamente, no que diz respeito à melhoria dos processos de negócio, redução de custos e transformação das organizações.

Um outro estudo conduzido pela consultora Accenture junto dos CIO e de outros executivos de topo, de organizações em todo o mundo e ainda um inquérito *online* a responsáveis pelo desenvolvimento de aplicações móveis, revela que dois terços (67%) dos CIOs e executivos de TI acreditam que a mobilidade vai ter tanto ou mais impacto nas suas empresas do que a Internet nos anos 90. O estudo conclui também que mais de dois terços (69%) dos inquiridos disponibilizariam mais de 20% do orçamento de investimento para a área de mobilidade, mostrando um contraste entre os responsáveis de TI nos mercados emergentes (94%) e dos mercados maduros (35%) (Accenture, 2012).

O mesmo estudo apurou, ainda, que 48% dos inquiridos nos mercados emergentes possuem uma estratégia de mobilidade amplamente desenvolvida, enquanto, apenas 12% dos inquiridos nos mercados maduros alegaram ter estratégia semelhante.

Segundo um artigo publicado na página web da Tek, datado de 28 Agosto de 2013, existem em Portugal 3.53 milhões de utilizadores de *smartphones*. Esse mesmo artigo refere que a percentagem de utilização passou de 25.8 % em maio de 2012 para 39.6% em agosto de 2013 (Tek, 2013).

1.2 Questões de Investigação

As principais questões de investigação deste trabalho são as seguintes:

- **Qual a relação existente entre e as Metodologias de Gestão de Projetos e o PMBOK?**

- **Como desenvolver uma aplicação Web e Mobile que disponibiliza de forma integrada o mapeamento entre as Metodologias de Gestão de Projetos e o PMBOK?**

1.3 Objetivo

Grande parte do trabalho realizado na indústria de desenvolvimento de sistemas de informação (SI) é realizada através de um projeto. Gerir um projeto é uma tarefa complexa, por isso, ter um conjunto de ferramentas que permitem a comunicação entre os múltiplos elementos da equipa e a escolha de uma metodologia adequada, são absolutamente fundamentais. Do ponto de vista organizacional são muitos os problemas que se levantam no ambiente de trabalho por projetos. Ficam aqui desde já enumerados alguns desses problemas:

- Gerir de uma forma coerente o projeto, desde a fase de caderno de encargos, proposta, Project Charter, Plano de Projeto, Desenvolvimento, Testes, Entrada em Produção e Manutenção;
- Planear o projeto, através da sua decomposição (WBS) e planeamento/calendarização.
- Integração com gestão de código fonte, e de bug tracking;
- Conhecer e definir quais os elementos que estão disponíveis para constituir equipas de trabalho, quais as suas competências e a sua disponibilidade temporal;
- Atribuição das diversas tarefas aos responsáveis pelas mesmas;
- Notificação e distribuição das tarefas e do trabalho pelos responsáveis da mesma. O sistema deve permitir visualizar essas tarefas na plataforma, assim como permitir importar as mesmas para os calendários pessoais;
- Implementação de estratégias GTD;
- Implementação de mecanismos de *reporting* do estado das tarefas;
- Visão integrada do Projeto em tempo real;
- Suporte para multi-projetos e multi-utilizadores.

Este trabalho visa por isso alcançar dois principais objetivos:

1. Fazer um mapeamento entre as principais metodologias de gestão de projetos (dando destaque às metodologias aplicadas no desenvolvimento de sistemas de

informação) e as áreas de conhecimento e grupos de processos de gestão de projeto definidos no PMBOK.

2. Desenvolver uma aplicação web que possa ser usada igualmente em dispositivos móveis disponibilizando toda a informação do primeiro objetivo, ou seja, nessa aplicação deve ser possível consultar, para uma determinada metodologia, que áreas de conhecimento e processos são contemplados.

1.4 Métodos de Investigação

A realização deste trabalho baseou-se essencialmente na pesquisa bibliográfica. No estado da arte realizou-se um pequeno estudo do paradigma da Gestão de Projetos e, também, o levantamento das metodologias e ferramentas de gestão de projetos. A pesquisa sustentou-se em bibliografias, artigos e conteúdos disponíveis na internet. Ainda, foi usada a pesquisa descritiva de modo a descrever as características das ferramentas de gestão de projetos como base para realização do estudo comparativo. Este tipo de pesquisa foi, também, usada na elaboração do mapeamento entre as metodologias de gestão de projetos e o PMBOK, em que, foi considerado os *standards* do Project Management Institute (PMI)², de modo a conferir a cobertura das metodologias às áreas de conhecimento e aos grupos de processos.

1.5 Organização do Documento

Este documento está organizado em seis capítulos:

No capítulo 1 (**Introdução**) apresenta-se a contextualização dos principais aspetos deste projeto de dissertação, a motivação e os objetivos do mesmo.

No capítulo 2 (**Estado da Arte**) apresenta-se o estado da arte, onde são apresentados os conceitos básicos de Gestão de Projeto, as principais metodologias de Gestão de Projeto e as ferramentas informáticas de apoio à Gestão de Projeto, dando destaque às suas principais funcionalidades, apresentando uma análise comparativa entre as mesmas.

No capítulo 3 (**Mapeamento do PMBOK com as Metodologias de Gestão de Projeto**) apresenta-se o mapeamento entre as áreas de conhecimento e processos de

² PMI é uma instituição internacional sem fins lucrativos que define os *standards* de gestão de projetos.

Gestão de Projeto definidos no PMBOK e as metodologias de Gestão de Projeto. Para cada metodologia será apresentado o mapeamento com áreas de conhecimentos e grupos de processos.

No capítulo 4 (**Processo de Implementação da Aplicação**) apresenta-se todo o processo de implementação da aplicação desenvolvida, incluindo a arquitetura, os requisitos, bem como as tecnologias utilizadas.

No capítulo 5 (**Validação da Aplicação**) apresenta-se a validação da aplicação desenvolvida. Para cada caso de uso serão identificados os requisitos que são validados por este.

No capítulo 6 (**Conclusão e Trabalho Futuro**) apresenta-se a conclusão do trabalho desenvolvido e as sugestões para o trabalho futuro.

2 Estado da Arte

2.1 Introdução à Gestão de Projetos

A Gestão de Projetos (GP) como ciência emergiu, progressivamente ao longo dos anos tendo surgido no início do século XX, nomeadamente na década de 1920 com a invenção do diagrama de Gantt (Silva, 2007).

Esta técnica foi criada pelo engenheiro americano Henry L. Gantt, que procurou resolver o problema da programação das atividades, através da sua distribuição num gráfico onde se podia visualizar rapidamente a duração, o início, o fim das atividades (Silva, 2007). Nesta altura reconheceu-se que a abordagem tradicional da gestão de projetos era insuficiente para dar resposta aos problemas surgidos na gestão de esforços únicos e inovadores.

Na segunda metade do século XX, verificou-se um acréscimo significativo na complexidade dos processos na área da indústria e serviços, necessitando de um apoio significativo da área de Gestão de Projeto.

Como resposta a esta necessidade, foram fundadas várias organizações profissionais dedicadas à área de Gestão de Projeto.

2.1.1 PMBOK Guide

O PMI (Project Management Institute) compilou as melhores práticas de gestão de projetos utilizadas no mundo, que são aplicadas em projetos de tamanhos e áreas diferentes, e criou uma publicação chamada **Project Management Body of Knowledge – PMBOK Guide**.

O PMBOK também fornece e promove um vocabulário comum dentro da profissão de gestão de projetos para se discutir, escrever e aplicar conceitos de gestão de projetos. Esse vocabulário padrão é um elemento essencial da profissão.

O PMI considera esta norma como uma referência básica de gestão de projetos para seus programas de desenvolvimento profissional e certificações. Como uma referência básica, esta norma não é abrangente nem completa. Ela é mais uma guia que uma metodologia (PMBOK, 2008).

2.1.2 Definição do Projeto

Segundo o PMI, um projeto é um empreendimento temporário levado a efeito com o objetivo de produzir um produto ou serviço único. Sendo o empreendimento uma sequência de atividades únicas, complexas e interligadas, que têm um objetivo ou propósito e que devem ser concluídas num determinado período de tempo, dentro de um dado orçamento e de acordo com uma certa especificação (Miguel, 2009).

Segundo esta definição é possível destacar as seguintes características de um projeto:

- **Sequência de Atividades** - um projeto compreende um conjunto de atividades que devem ser realizadas numa determinada sequência.
- **Atividades Únicas** - as circunstâncias de um determinado projeto nunca aconteceram antes e não sucederão mais.
- **Atividades Interligadas** - a interligação implica a existência de relações lógicas ou técnicas entre atividades. Existe uma ordem para a sequência de realização das atividades que constituem o projeto. Elas são consideradas interligadas porque o *output* de uma atividade é o *input* de outra.
- **Um Objetivo** - os projetos podem ter um único objetivo.
- **Produto ou Serviço Único** - o produto ou serviço alvo do projeto pode ser único, embora a categoria que pertence seja ampla.
- **Num Dado Período de Tempo** - os projetos têm uma data de conclusão, que pode ser imposta internamente pela gestão ou externamente por um cliente.
- **Com um Dado Orçamento** - os projetos têm sempre recursos limitados – pessoas, dinheiro, equipamentos, etc.
- **De acordo com uma Especificação** - o cliente ou o recetor dos resultados do projeto, espera um certo nível de funcionalidade e qualidade para o projeto.

2.1.3 Gestão de Projetos (GP)

Gestão de Projetos é a aplicação do conhecimento, habilidades, ferramentas e técnicas às atividades do projeto a fim de atender aos seus requisitos (PMBOK *Guide* 2008).

Para Kerzner (2005) a Gestão de Projetos consiste no planeamento, organização, direção e controlo dos recursos de uma empresa para um objetivo de relativamente curto prazo relativo que foi estabelecido para a concretização de objetivos específicos. Além

disso, a gestão de projetos utiliza a abordagem sistémica à gestão de forma a alocar o pessoal funcional (hierarquia vertical) a projetos específicos (hierarquia horizontal).

Roldão (2005) define Gestão de Projetos como processo de planeamento execução e controlo de um projeto, desde o seu início até à sua conclusão, com vista à consecução de um objetivo final num certo prazo, com um certo custo e qualidade, através da mobilização de recursos técnicos e humanos.

A Gestão de Projetos é compreendida em cinco grupos processos (Figura 1) (PMBOK, 2008):

- **Grupo de Processos de Iniciação** – processos realizados com o objetivo de definir e autorizar um novo projeto ou uma fase de um projeto;
- **Grupo de Processos de Planeamento** – processos realizados com o objetivo de definir o âmbito do projeto, refinar objetivos e definir o curso de ação necessário para alcançar os objetivos e o âmbito para que o projeto foi iniciado;
- **Grupo de Processos de Execução** – processos realizados com o objetivo de integrar pessoas e outros recursos para executar o trabalho definido no plano do projeto;
- **Grupo de Processos de Monitorização e Controlo** – processos realizados com o objetivo de monitorizar, rever e regular o progresso e o desempenho do projeto, identificar áreas em que seja necessário efetuar alterações ao plano de projeto e executar essas alterações;
- **Grupo de Processos de Encerramento** – processos realizados com o objetivo de concluir todas as atividades ao longo de todos os grupos de processo para encerrar formalmente o projeto ou uma fase do projeto.

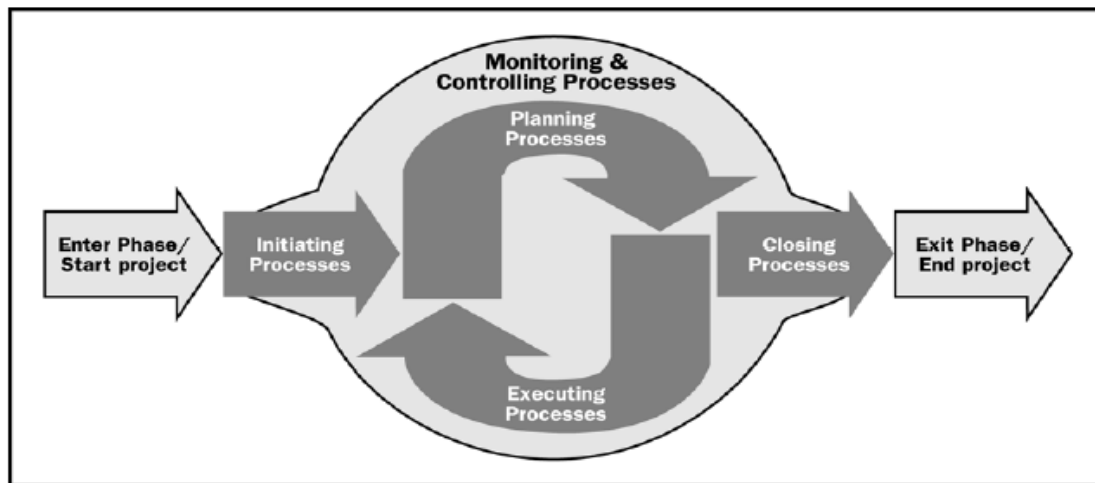


Figura 1: Grupos de processos da Gestão de Projetos. Fonte: (PMBOK)

Os processos de Gestão de Projetos estão organizados em nove áreas do conhecimento (Figura 2) (PMBOK, 2008).

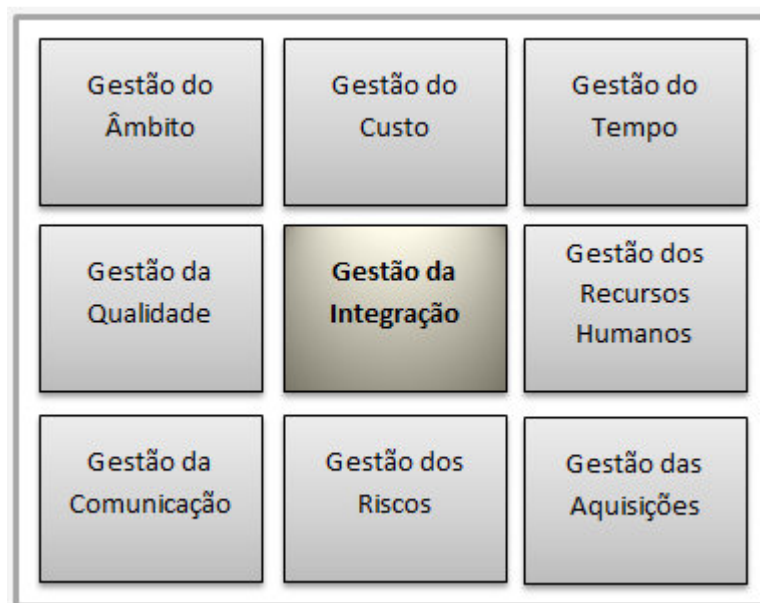


Figura 2: Áreas de conhecimento da Gestão de Projetos

ÁREAS DE CONHECIMENTO	GRUPOS DE PROCESSOS				
	Início	Planeamento	Execução	Monitorização e Controlo	Encerramento

4. Gestão da Integração	4.1 Desenvolvimento do Project Charter	4.2 Desenvolvimento do Plano de Gestão de Projeto	4.3 Direção e Gestão da Execução do Projeto	4.4 Monitorização e Controlo do Trabalho do Projeto 4.5 Controlo Integrado da Mudança	4.6 Encerramento do Projeto ou Fase
5. Gestão do Âmbito		5.1 Recolher os Requisitos 5.2 Definir o Âmbito 5.3 Criar o WBS		5.4 Verificar o Âmbito 5.5 Controlar o Âmbito	
6. Gestão do Tempo		6.1 Definir Atividades 6.2 Sequenciar Atividades 6.3 Estimar Recursos das Atividades 6.4 Estimar Duração das Atividades 6.5 Criar Calendarização		6.6 Controlar a Calendarização	
7. Gestão do Custo		7.1 Estimar Custos 7.2 Determinar o Orçamento		7.3 Controlar Custos	
8. Gestão da Qualidade		8.1 Planear a Qualidade	8.2 Realizar a Verificação da Qualidade	8.3 Controlar a Qualidade	
9. Gestão de Recursos Humanos		9.1 Desenvolver o Plano de Recursos Humanos	9.2 Constituir a Equipa de Projetos 9.3 Desenvolver a Equipa de Projeto 9.4 Gerir a Equipa de Projeto		
10. Gestão da Comunicação	10.1 Identificar os Stakeholders	10.2 Planear a Comunicação	10.3 Distribuir Informação 10.4 Gerir as Expectativas dos Stakeholders	10.5 Relatar o Desempenho	
11. Gestão dos Riscos		11.1 Planear a Gestão de Risco 11.2 Identificar Riscos 11.3 Análise Qualitativa de Risco 11.4 Análise Quantitativa de Risco 11.5 Planear Respostas a Risco		11.6 Monitorar os Riscos	
12. Gestão das Aquisições		12.1 Planear Aquisições	12.2 Conduzir Aquisições	12.3 Gerir Aquisições	12.4 Fechar Aquisições

Tabela 1: Mapeamento de grupos de processos e áreas de conhecimento da Gestão de Projetos. Fonte: (Adaptado do PMBOK)

A tabela anterior (Tabela 1) faz uma ligação entre processos de gestão de projetos com as áreas de conhecimento da gestão de projetos e permite verificar em que processo se usam determinados conhecimentos.

2.1.4 Ciclo de Vida do Projeto

Segundo o PMBOK Guide o ciclo de vida de um projeto consiste nas fases do mesmo, que geralmente são sequenciais e que às vezes se sobrepõem, cujo nome e número são determinados pela gestão e de acordo com controlo da(s) organização(ões) envolvida(s), a natureza do próprio projeto e a área de aplicação do projeto.

Embora os projetos possam variar substancialmente em dimensão e complexidade, todos eles podem ser decompostos de acordo com a seguinte estrutura básica de ciclo de vida:

- Iniciar o projeto;
- Organizar e planear o projeto;
- Executar o trabalho do projeto;
- Encerrar o projeto.

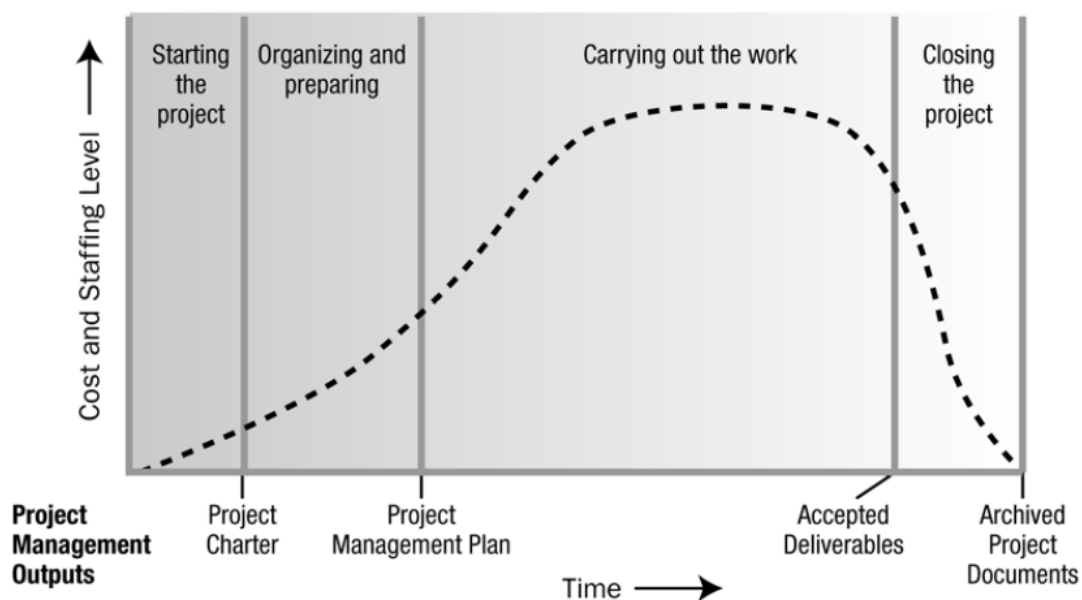


Figura 3: Fases do ciclo de vida de um projeto. Fonte: (PMBOK)

2.2 Metodologias de Gestão de Projetos de SI

O grande desafio que é o desenvolvimento de um projeto de forma eficaz e que satisfaça os interesses do cliente, e que é cada vez mais crescente com as exigências dos mesmos e do mercado em geral, fez com que ao longo dos tempos as empresas sentissem a necessidade de criar processos de auxílio à gestão e implementação de projetos devido aos problemas resultantes do modo como os mesmos são geridos. Por forma a resolver ou minimizar tais problemas foram surgindo metodologias e processos aplicáveis na gestão de projetos.

Segundo Charvat (2003), a metodologia é um conjunto de orientações ou princípios que podem ser adaptados e aplicados a uma situação específica. Num ambiente de projeto essas diretrizes podem ser uma lista de coisas para fazer. A metodologia também pode ser uma abordagem específica, modelos, formas e mesmo *checklists* utilizados ao longo do ciclo de vida do projeto.

Conforme apresenta a tabela seguinte (Tabela 2), Charvat (2003) faz a distinção entre metodologias de GP e metodologias de desenvolvimento. Para ele, embora estas duas metodologias andam de mãos dadas, existem diferenças. Considera que as metodologias de GP estabelecem um *framework* de alto nível de Gestão de Projetos, enquanto, as metodologias de desenvolvimento fornecem o detalhe do desenho do sistema e desenvolvimento.

Description	Suited to control of:				Phases	Project Size	Comments
	S	Q	T	\$			
Project Management Frameworks Methodologies							
Rational Unified Process	Y	Y	Y	Y	Y	M, L	1, 2, 3, 4
PRINCE2	Y	Y	Y	Y	Y	M, L	4
System Development Life Cycle (SDLC)	Y	Y	N	?	Y	S, M, L	3, 4, 6
Solutions-based Project Methodology	Y	Y	N	N	Y	S, M	3, 5
TenStep	Y	Y	Y	N	N	S, M	5
Technology Development Management Methodologies							
The "Agile" Group:							
Extreme Programming (XP)	N	Y	N	N	N	S, M	5
Scrum	N	Y	N	N	N	S, M	5
Crystal	N	Y	N	N	N	S, M	5, 7
Dynamic Sys. Development (DSDM)	Y	Y	Y	?	Y	S, M	5
Rapid Applications Development (RAD)	Y	Y	Y	?	Y	M, L	5
Unicycle	Y	Y	Y	Y	Y	S, M, L	4
Code-and-fix Approach	N	N	N	N	N	S	7
V-methodology	Y	Y	Y	Y	Y	M, L	4
Waterfall	Y	Y	Y	Y	Y	M, L	4, 6
Open Source	N	N	N	N	N	S, M	5
Spiral	Y	Y	N	N	Y	M, L	4
Synchronize and Stabilize	Y	Y	N	N	Y	M, L	
Reverse Engineering Development	Y	Y	N	N	Y	M, L	4
General Publication Methodology	Y	Y	N	?	Y	M	4, 8
Structured System Analysis & Design	Y	Y	N	N	Y	M, L	4
Pramis	Y	Y	Y	Y	Y	M, L	4
Offshore Development	Y	Y	Y	Y	Y	L	4
General Drug Development	N	Y	N	N	Y	L	4
Classic Building Construction	Y	?	Y	Y	Y	M, L	4

Tabela 2: Comparison of various methodologies from a project management perspective.

Fonte: (Charvat, 2003)

Legenda:

- ***Suited to control of:*** S: Scope, Q: Quality, T: Time, \$: Cost
- ***Comments:***
 1. Y, N, ?: Yes, No, Undetermined
 2. S, M, L: Small, Medium or Large projects
 3. Arguably an IT/software development methodology, i.e. belongs under Technology Management
 4. High management ceremony
 5. Low management ceremony
 6. Classic "waterfall" sequence
 7. Not suited to virtual teams
 8. For book and periodical publishing

Seguidamente são analisadas algumas das metodologias apresentadas na Tabela 2.

2.2.1 Rational Unified Process (RUP)

O RUP é um processo de engenharia de software criada pela Rational Software Corporation, que foi adquirida pela IBM. Uma das principais diferenças entre RUP e outras metodologias é que o RUP não usa uma abordagem em cascata para o desenvolvimento de SI. As fases de requisitos, análise, projeto, implementação, integração e os testes não são feitos em sequência estrita. No RUP, é usada uma abordagem iterativa (Figura 5), onde um produto de software é projetado e construído numa sucessão incremental de iterações. Cada iteração inclui algumas, ou mais, das disciplinas de desenvolvimento (requisitos, análise, projeto, implementação, teste e assim por diante).

É um processo aplicado preferencialmente a grandes equipas de desenvolvimento e a grandes projetos, uma vez que é amplamente adaptável. Devido à natureza modular do RUP, ele pode ser utilizado para todos os tipos de projetos de software. No entanto, por causa da complexidade da metodologia RUP, é utilizado principalmente para grandes projetos de software.

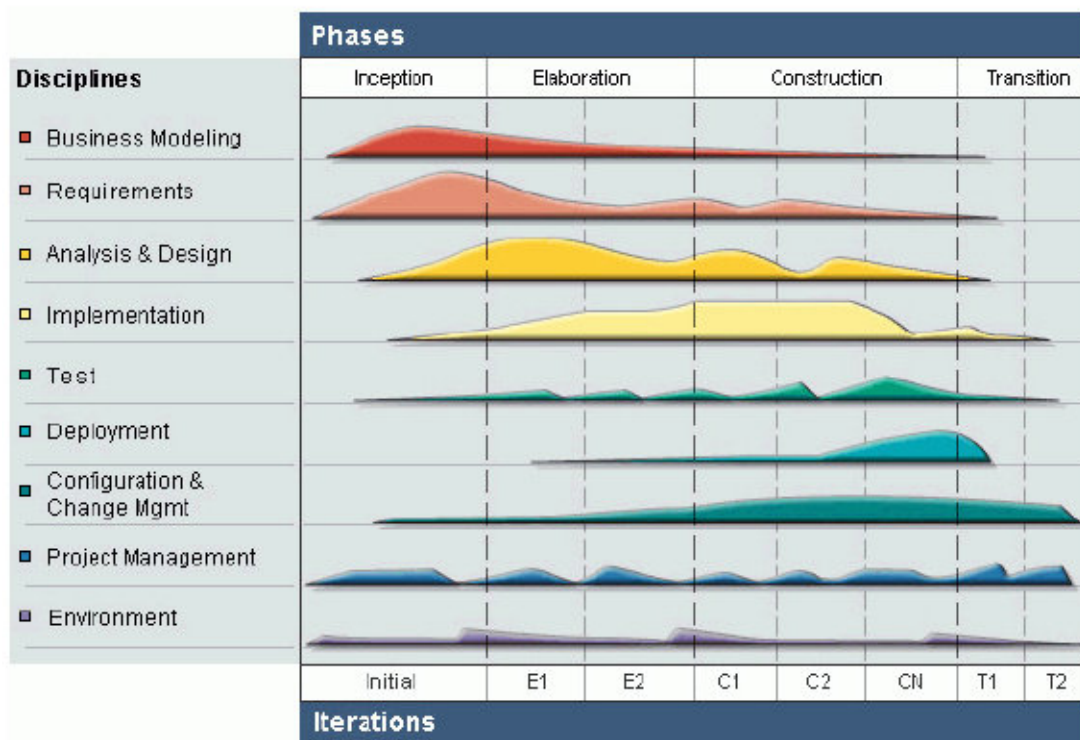


Figura 4: RUP. Fonte: (IBM, 2012)

O processo iterativo do RUP é composto por quatro fases (Figura 4):

- ***Inception (What to build)*** - âmbito do projeto, requisitos de alto nível, estimativa de recursos.
- ***Elaboration (How to build it)*** - detalhe nos requisitos, produção de uma arquitetura estável de acordo com casos de uso.
- ***Construction (Build the product)*** - completar os requisitos e desenhar o modelo, implementar e testar cada componente, recorrendo a protótipos nos utilizadores finais, lançar as primeiras versões do software funcional.
- ***Transaction (Deploy to end users)*** - produzir versões incrementalmente com base nos erros corrigidos, atualizar os manuais a cada versão.

Cada fase é composta por várias iterações representadas esquematicamente na figura seguinte (Figura 5):

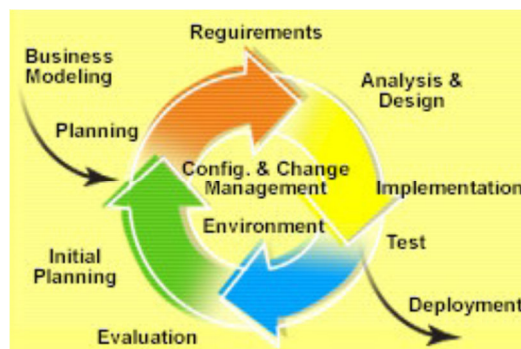


Figura 5: Fluxo de iterações do RUP. Fonte: (IBM, 2012)

2.2.2 PRINCE2 (Projects in Controlled Environments – 2ª versão)

É um método baseado em processos para a gestão eficaz de projetos. Foi originalmente desenvolvida pela Central Computer and Telecommunications Agency (CCTA) agora faz parte do Office of Government Commerce (OGC). Desde 1989 tem sido utilizado como um padrão para gestão de projetos especialmente no Reino Unido (PRINCE2, 2012a). Este método foi desenvolvido inicialmente apenas para os projetos de SI, esta versão é consistente com a gestão de todos os tipos de projetos.

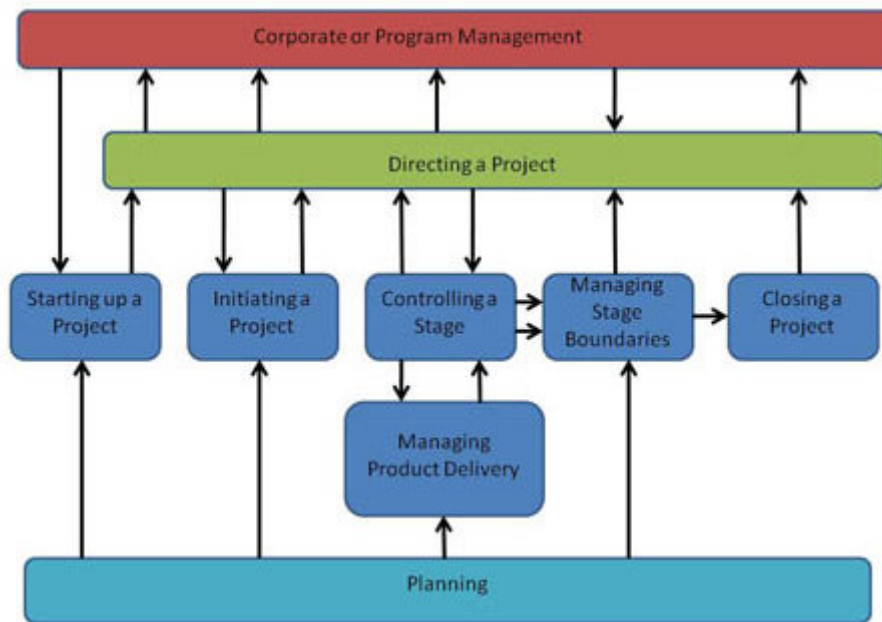


Figura 6: Processos do PRINCE2. Fonte: (PRINCE2, 2012b)

Os processos do PRINCE2 estão apresentados na figura anterior (Figura 6).

- **Gerir o Projeto (*Directing a Project*)** - a gestão de projetos é executada do princípio ao fim. Os processos chave da direção do projeto estão divididos em 4 fases:
 - Iniciação (iniciar o projeto com o pé direito)
 - Limites estágio (compromisso de mais recursos após ter verificado resultados);
 - Direção *ad hoc* (*Ad hoc Direction*) (controlar o progresso, fornecendo orientação e conselhos, reagindo às situações inesperadas);
 - Fecho do projeto (confirmar o resultado e controlar o fecho);
- **Arranque do Projeto (*Starting Up a Project*)** – este é o 1º processo do PRINCE. É um processo pré-projeto, designado para assegurar que os pré-requisitos para iniciar o projeto estão bem definidos. É esperado neste processo que esteja definido a alto nível a razão do projeto e o que é se pode esperar do mesmo. Este processo deve ser muito curto.
- **Iniciar o Projeto (*Initiating a Project*)** - os objetivos de iniciar um projeto são:
 - Concordar se há ou não justificação para prosseguir;
 - Estabelecer uma base estável de gestão para prosseguir;

- Documentar e confirmar que existe um exemplo aceitável do negócio para o projeto;
 - Assegurar que a empresa cliente procedeu à aceitação antes de começar o trabalho;
 - Concordar com o compromisso dos recursos para a primeira etapa;
 - Permitir incentivar o quadro do projeto a tomar posse do projeto;
 - Fornecer uma linha de base para os processos de tomada de decisão requeridos durante o ciclo de vida do projeto;
 - Certificar que o investimento do tempo e esforço requerido pelo projeto seja feito sabiamente, tendo em conta os riscos a ele inerentes.
- **Controlar os Limites da Etapa (*Managing a Stage Boundary*)** – com este processo a direção do projeto sabe se deve ou não continuar com o projeto. Os objetivos do processo são os seguintes:
- Assegurar que no plano corrente todos os resultados foram completados conforme definidos;
 - Providenciar a informação necessária ao plano para avaliar a continuidade do projeto;
 - Providenciar informação necessária no plano para aprovar a etapa como completa e autorizar o início da próxima etapa;
 - Registar algumas medidas e lições que possam ajudar etapas em atraso do projeto e/ou outros projetos.
- **Controlar uma Etapa (*Controlling a Stage*)** - este processo descreve o controlo de atividade para assegurar que a etapa se mantém em curso e reaja a eventos inesperados. Em toda a etapa haverá um ciclo que consiste em:
- Trabalho autorizado a ser feito;
 - Reuniões de progresso com informação sobre o trabalho;
 - Verificar as alterações;
 - Revisões da situação;
 - Relatórios;
 - Fazer correções necessárias;

Para além destas atividades, é feita também a gestão do risco e o controlo de alterações.

- **Gestão de Entrega do Produto (*Managing Product Delivery*)** - o objetivo deste processo é ter a certeza que o produto que foi planeado é criado e entregue por:
 - Ter a certeza que o trabalho afeto à equipa é efetivamente autorizado, realizado, aceite e verificado;
 - Ter a certeza que o trabalho está de acordo com os requisitos identificados inicialmente;
 - Garantir que os produtos terminados atendem aos critérios de qualidade;
 - Controlar o progresso do trabalho e ter uma previsão regular.
- **Fecho do Projeto (*Closing a Project*)** - o objetivo deste processo é executar o controlo do fecho do projeto. Os objetivos do processo de encerramento do projeto são:
 - Verificar se os objetivos iniciais foram cumpridos e verificar os que foram feitos a mais;
 - Verificar a satisfação do cliente;
 - Assegurar que todas as extensões do produto foram terminadas e aceites pelo cliente;
 - Deixar algumas recomendações das ações a seguir;
 - Analisar as lições aprendidas com o projeto e completar o relatório das lições aprendidas;
 - Preparar o relatório final do projeto;
 - Notificar a pessoa responsável da organização da intenção de dissolver a organização do projeto e os recursos.
- **Planeamento (*Planning*)** – nesta fase é feito a estimativa do esforço requerido pelas atividades que dão origem ao produto e calendarizam-se todas as atividades num plano. Também nesta fase deve ser feita a análise de riscos e os critérios de fim de cada fase (*milestone*) e do projeto como um todo.

2.2.3 TenStep

TenStep é uma metodologia de Gestão de Projetos que foi concebida de acordo com um conjunto de princípios orientadores (TenStep, 2012a).

Apresenta como principais características:

- **Flexível e escalável** – um processo de gestão de projetos deve ser flexível e escalável, com base no tamanho do respetivo projeto. Na metodologia Tenstep, este conceito é referido como “pequenas metodologias para projetos pequenos, grandes metodologias para projetos grandes”. A escalabilidade está relacionada com o nível de complexidade dos processos de gestão de projetos, assim como o tempo e a dedicação despendidos.
- **Aplicável a todos os tipos de projetos** – a metodologia Tenstep foi concebida para ser aplicável a todos os projetos, quer seja na construção de uma casa, um circuito integrado ou uma aplicação informática.
- **Gestão pró-ativa** – os projetos devem ser geridas pro-ativamente, independentemente da sua dimensão. O gestor de projetos que espera que as coisas aconteçam, irá certamente deparar-se com problemas.
- **Estabelecer processos e parcerias entre a equipa do projeto e o cliente** – os processos que serão utilizados na gestão do projeto devem ser estabelecidos logo no início e compreendidos pela equipa e o cliente. Grande parte dos processos requer o envolvimento de elementos do cliente e da equipa do projeto. Estes elementos não irão compreender o seu papel nesses processos se estes não foram previamente discutidos.
- **Os gestores de projetos devem ter um nível suficiente de autoridade** – se o gestor de projeto é responsável pela entrega do projeto e, no entanto, não puder tomar decisões chave necessárias para gerir, então provavelmente terá dificuldades em garantir o sucesso do projeto.

Os 10 passos da metodologia TenStep (Figura 7) não implicam uma ordem sequencial (TenStep, 2012b). Estes passos não têm que ser todos executados e de forma sequencial. Os passos 1 e 2 são os primeiros a ser executados, uma vez que para gerir algo, é preciso definir e planear o mesmo. No entanto, as atividades aplicáveis no passo 3 até o passo 10 são realizadas em paralelo.

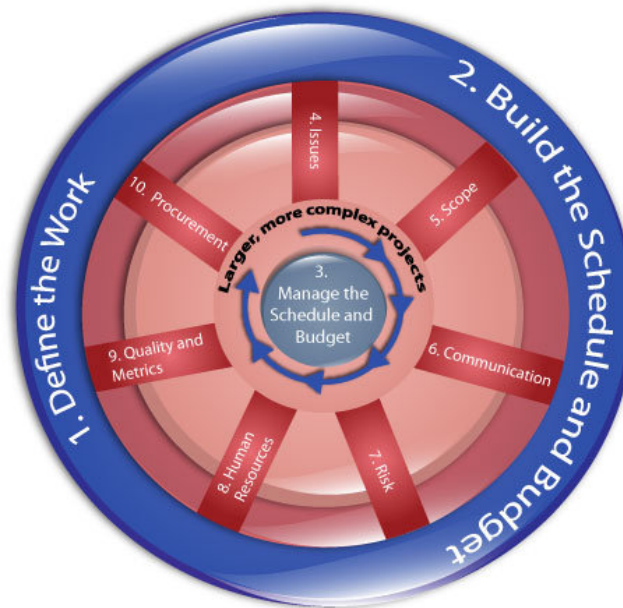


Figura 7: Passos da metodologia TenStep. Fonte: (TenStep, 2012b)

2.2.4 Extreme Programming (XP)

XP é uma metodologia de desenvolvimento ágil de software, fundada por Kent Beck ³ nos meados da década de 90, com o principal enfoque na redução de custos de mudança. Esta metodologia funciona baseado nos valores como a comunicação, simplicidade, *feedback* e coragem (Beck, 2000).

- **Comunicação** – os programadores XP comunicam com os seus clientes e com os seus colegas programadores;
- **Simplicidade** - esta metodologia procura encontrar o caminho mais simples a seguir em qualquer situação. Requisitos devem ser simples, o programa deve ser simples e a disciplina em si deve ser simples. A simplicidade facilita a comunicação;
- **Feedback** – obter feedback do cliente desde o 1º dia do projeto. Entregar o produto ao cliente o mais cedo possível e posteriormente fazem as alterações sugeridas;
- **Coragem** - com estes fundamentos os programadores XP podem responder corajosamente às mudanças nos requisitos e na tecnologia.

³ Engenheiro de software Americano, criador da Extreme Programming e de metodologias de desenvolvimento de software orientado a teste conhecido como desenvolvimento ágil de software.

No XP cada colaborador do projeto é parte integrante da equipa (*Whole Team*), e o cliente também faz parte dessa equipa. As histórias dos clientes são escritas. No XP foram definidos um conjunto de boas práticas que tornam a metodologia ágil e simples de usar (Beck, 2000).

O XP é uma metodologia aplicada a pequenas e médias equipas de desenvolvimento de software, utilizado quando os requisitos não são claros e/ou sofrem muitas alterações ao longo do desenvolvimento, onde é aconselhável que a equipa deve ter no máximo 10 pessoas, podendo até ser superior, mas convém não chegar a 20, que seria demais, segundo Beck (Charvat, 2003).

2.2.5 SCRUM

Scrum é uma metodologia de gestão ágil de projetos, baseada numa abordagem iterativa e incremental de desenvolvimento de software.

É uma estrutura que tem sido usada na gestão do desenvolvimento de produtos complexos desde 1990; Não é um processo ou uma técnica para desenvolver produtos, mas sim uma estrutura em que se pode empregar vários processos e técnicas (Schwaber e Sutherland, 2011).

O Scrum é composto por Equipas Scrum e as suas funções, por **eventos**, **artefactos** e **regras**. Cada componente dentro desta estrutura serve um propósito específico e é essencial ao uso e sucesso do Scrum (Schwaber e Sutherland, 2011).

A Figura 8 apresenta os principais conceitos do Scrum:

- **Product Backlog** - é uma lista ordenada de tudo o que possa ser necessário no produto e é a única fonte de requisitos para as alterações a serem feitas ao produto.
- **Sprint Backlog** - é o conjunto de itens do *Product Backlog* que foram selecionados para o *Sprint*, mais o plano para entregar um incremento do produto e atingir a meta da *Sprint*. O *Sprint Backlog* é uma previsão da equipa de desenvolvimento sobre que funcionalidade estará no próximo incremento e o trabalho necessário para entregar essa funcionalidade.

- ***Sprint*** - é o coração do Scrum, uma caixa de tempo de um mês ou menos durante o qual um “*Done*”, que é utilizável e é um incremento de produto potencialmente comercializável, é criado. Os *Sprints* possuem duração consistente durante todo o desenvolvimento. O seu planeamento é feito através do *Sprint Planning Meeting*, onde participam todos os interessados, e é definido o *Sprint Backlog*. O *Sprint* novo começa imediatamente após a conclusão do *Sprint* anterior.

Durante o *Sprint*, a equipa encontra-se diariamente durante 15 minutos numa reunião designada de *Daily Scrum*, de forma a controlar o progresso do projeto e identificar possíveis dificuldades. No final de cada *Sprint*, uma versão do produto com novas funcionalidades deve estar concluída. As tarefas que ficaram por concluir retornam ao *Product Backlog*. Antes do início de um novo *Sprint* é realizada uma reunião, a *Sprint Review Meeting*, na qual a equipa apresenta ao *Product Owner* as funcionalidades implementadas no último *Sprint*. O *Product Owner* é convidado a pronunciar-se sobre possíveis alterações, que serão consideradas em futuros *Sprints*.

Por fim, existe uma reunião entre a *Scrum Team* e o *Scrum*, designada de *Sprint Retrospective*. Esta reunião serve para analisar a aplicação do processo Scrum, promovendo a discussão de melhores práticas e formas de tornar o processo mais eficiente.

O Scrum em si não diz como um produto deve ser projetado, neste sentido pode ser aplicado juntamente com outros processos mais ligados à prática de engenharias. Neste caso o XP (Silva e Videira, 2008).

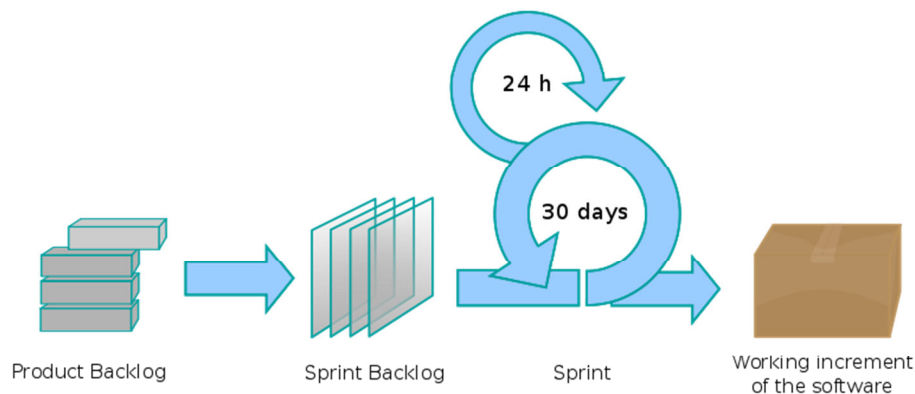


Figura 8: Processo Scrum. Fonte: (WKP)

2.3 Ferramentas de Gestão de Projetos (FGP)

Atualmente existem inúmeras ferramentas de auxílio à Gestão de Projetos que podem ajudar os gestores de projeto a desempenharem as suas tarefas.

As ferramentas de Gestão de Projetos podem ser definidas como software que ajuda o gestor de projetos a planejar, acompanhar e controlar a execução de um projeto. Independentemente da complexidade do projeto, fazer alterações e atualizar o cronograma à medida que o projeto se desenvolve, é algo que não se consegue fazer sem o uso de ferramentas informáticas, uma vez que uma pequena alteração pode implicar a modificação de todo o cronograma.

Uma ferramenta de apoio à Gestão de Projetos, em geral, deve ser capaz de, fornecer a informação certa, no formato certo, às pessoas certas e no tempo certo, para que as melhores decisões possam ser tomadas.

Nesta análise, as ferramentas de Gestão de Projetos serão categorizadas em termos da disponibilidade em termos de plataforma, como ferramentas desktop e ferramentas Web (*Web-Based*), e dentro destes grupos em ferramentas específicas para GP de SI, que suportam as metodologias ágeis de desenvolvimento de software.

2.3.1 Análise das Ferramentas Desktop

Entende-se como ferramentas desktop de GP, softwares que funcionam instalados no próprio computador.

2.3.1.1 Microsoft Project (Ms Project)

O Ms Project é um software de GP desenvolvido pela Microsoft. A primeira versão foi lançada em 1985. É uma ferramenta desktop eficaz e flexível, com uma interface gráfica amigável que tem sofrido grandes melhorias e dispondo de poderosos recursos que permitem a gestão de projetos em diferentes áreas, sejam eles simples ou complexos.

Funciona integrado com todos os outros produtos da família Office, facilitando assim a interligação dos diversos tipos de trabalho numa organização (Microsoft, 2013).

Apresenta como principais funcionalidades:

- Novo visual gráfico;

- Planeamento das atividades do projeto: propriedade, calendário, características, sequência, duração e caminho crítico;
- Gestão de recursos: definição dos recursos (pessoas, equipamentos e materiais) e configuração das suas informações e afetação a atividades;
- Controlo de sobreafetação de recurso;
- Gestão de custos: definição dos custos fixos e variáveis de acordo com os recursos;
- Rastrear caminho das tarefas, relatórios com gráficos *burndown*;
- Técnicas PERT e CPM na gestão de risco;
- Técnica EVM no controlo de prazos e custos;
- Multi-projeto, com possibilidade de criar sub-projetos;
- Importação/exportação de dados, guardar e partilhar ficheiros na nuvem.

2.3.1.2 OpenProj

OpenProj é um software de gestão de projetos *open source*, muito utilizado em alternativa ao MS Project. É multi-plataforma e funciona em qualquer sistema operativo, tem uma interface amigável e muito similar ao MS Project. Abre arquivos do MS Project mas possui um formato próprio de arquivo (Smashingapps, 2013).

Apresenta como principais funcionalidades:

- Visualização de gráficos Gantt e PERT;
- Visualização da EAP com duração das fases e custos das atividades;
- Gestão de recursos e custos associados;
- *Baseline* de projeto para acompanhamento da execução;
- *Earned value management*.

2.3.1.3 Gantt Project

Gantt Project é um software *open source* de gestão de projetos. Foi desenvolvido em Java e funciona no Windows, Linux e Mac OSX. É uma ferramenta muito simples que faz a gestão de projetos através do cronograma das tarefas e gráfico de Gantt.

Apresenta como principais funcionalidades (Smashingapps, 2013):

- Gestão de tarefas e suas dependências;
- Gestão e alocação de recursos humanos;

- Relatórios e gráfico de Gantt, com possibilidade de exportar em vários formatos (imagem, html e pdf);
- Importação/exportação de/para Microsoft Project;
- Partilha de projetos através de WebDav.

2.3.1.4 Open Workbench

É uma aplicação desktop gratuita de gestão de projetos que funciona em Windows, compatível com o MS-Project, sem limitações de utilizadores, permite um planeamento e controlo do projeto incluindo sub-projetos e todas as suas dependências (OpenWorkbench, 2013).

Apresenta como principais funcionalidades:

- Gestão de todos os recursos do projeto;
- Relatórios e gráficos de Gantt;
- Importação de dados para MS-Project;
- Gestão do plano do projeto.

2.3.2 Análise das Ferramentas Baseadas na Web (*Web-Based*)

Entende-se como ferramentas baseadas na Web, software que funciona *online*, ou seja, acessíveis através da internet. Algumas dessas ferramentas podem ser instaladas num servidor local.

2.3.2.1 Basecamp

É um sistema web colaborativo de gestão de projetos desenvolvido pela 37Signals, muito utilizado por pequenas e médias empresas. Possui uma interface simples de usar, oferecendo uma visão extremamente simplificada das etapas do projeto. Mas falha ao nível de planeamento e controlo, apresentado apenas as *to-do lists* com marcações de *milestones*, ou seja, não existe o conceito de tarefa, o que torna impossível criar dependências e alocar recursos (Basecamp, 2013).

Apresenta como principais funcionalidades:

- Forte componente de comunicação, disponibilizando áreas de trabalho para potenciar a criatividade das equipas e ainda um *chat* interno;
- Suporte multi-projeto nas versões comerciais;
- Gestão fácil de perfis de acesso aos diferentes projetos;

- Gestão de documentos calendários e relatórios;
- Multi-lingua e suporte *online* aos utilizadores.

2.3.2.2 Redmine

É um sistema de gestão de projetos *open source* baseado na web. Foi desenvolvido usando a *framework Ruby-on-Rails*, disponibilizado sob a licença GPL. Pode ser configurado para correr em várias plataformas e suporta múltiplas bases de dados (Redmine, 2013).

Apresenta como principais funcionalidades (Redmine, 2013):

- Suporte a múltiplos projetos;
- Níveis flexíveis de acesso e controlo;
- Gráfico de Gantt e calendário;
- Gestão de notícias e documentos;
- Gestão flexível da estrutura de projeto, criação de tarefas e sub-projetos como parte de um projeto;
- Notificações por email e RSS *feed*;
- Integração com sistemas de controlo de versões;
- Personalização de campos, autenticação LDAP;
- Múltiplas bases de dados e multi-lingua.

2.3.2.3 dotProject

É um sistema de gestão de projetos *open souce* baseado na web (pode ser instalado num servidor local). Começou a ser desenvolvido no ano 2000, usa a linguagem PHP e a base de dados MySQL (DotProject 2013a; DotProject 2013b) e é disponibilizado sob a licença GNU- GPL. Combina o planeamento, gestão e colaboração no projeto numa interface simples, onde é possível definir empresas, utilizadores e projeto. As suas funcionalidades podem ser estendidas (é um *framework*).

Apresenta como principais funcionalidades (DotProject 2013a; DotProject 2013b)

- Criação e gestão de empresas/companhia, onde posteriormente são atribuídos os projetos;
- Gestão de tarefas: dependências, prioridade, níveis de acesso, custo e estado;

- Suporte a multi-projeto,
- Gráfico de Gantt, calendário e relatórios;
- Gestão de recursos, arquivos e *tickets*;
- Ferramentas de administração;
- Fóruns de discussão, gestão de utilizadores, contactos e eventos;
- Importação/exportação de dados.

2.3.2.4 Gantt

Gantt é um software de GP disponível na web, ideal para gestão de recurso e cronograma do projeto. Faz uma gestão colaborativa onde os dados são atualizados e partilhados em tempo real. As tarefas são organizadas no gráfico de Gantt fazendo *drag-and-drop*.

Apresenta como principais funcionalidades (Gantt, 2013):

- Customização dos campos de dados de projetos e tarefas;
- Colaboração da equipa de projeto em tempo real;
- Gestão de recursos por equipa;
- Integração com calendário do Google;
- Registo automático do histórico de alterações;
- Multi-projeto e multi-utilizador;
- Relatórios customizáveis.

2.3.2.5 DO.Com

É uma plataforma colaborativa de gestão de projetos, que pode ser usado na web ou nativamente no IOS ou Android. Inicialmente foi lançada com o nome de Manymoon e após ser adquirida pela empresa Salesforce.com, foi relançada como DO.Com sofrendo uma grande alteração. Está disponível para instalação no Google *play* (DO.Com, 2013).

Apresenta como principais funcionalidades:

- Reutilização de projetos e lista de tarefas;
- Gestão colaborativa: gestão de grupos, conversação, email e alertas automáticas;
- Gestão de *templates*, calendários e documentos;
- Integração com serviços do Google;
- Acesso através de plataformas IOS e Android.

2.3.2.6 AceProject

É uma plataforma colaborativa de gestão de projetos na web e em ambientes móveis. É multi-projeto e cada projeto pode ser configurado com a sua própria estrutura, permitindo a gestão de diferentes tipos de projetos de acordo com as suas necessidades. Além disso, modelo de um projeto pode ser reutilizado na criação de outros projetos, caso estes sejam semelhantes ao anterior, evitando a configuração dos mesmos dados novamente (AceProject, 2013). Todos os seus recursos estão otimizados para dispositivos móveis.

Apresenta como principais funcionalidades (AceProject, 2013):

- Gestão de tarefas, dependência entre tarefas, portefólios e *template* do projeto;
- Relatórios, calendários e gráficos de Gantt;
- Gestão colaborativa: notificações de email, caixa de correio interno, fórum de discussão, caracteres internacionais e lembretes de tarefas;
- Gestão de recursos humanos: privilégios, grupos, preferências, estatísticas, etc;
- Gestão de documentos: partilha, controlo de versões e bloqueio;
- Gestão de *timesheet*: tempo e custo;
- Multi-browser, importação/exportação de dados.

2.3.2.1 Análise das Ferramentas de Suporte ao Desenvolvimento Ágil

Entendem-se como ferramentas de desenvolvimento ágil as ferramentas que suportam as metodologias de desenvolvimento ágil de software (explicado mais à frente no Capítulo 3).

2.3.2.1 TargetProcess

É uma ferramenta comercial de gestão ágil de projetos na web, desenvolvida pela TargetProcess, Inc. Suporta metodologias ágeis de desenvolvimento como SCRUM e XP. Pode ser adaptada de forma a responder às necessidades do processo de desenvolvimento escolhido (TargetProcess, 2013).

Apresenta como principais funcionalidades:

- Criação de metodologias personalizada para gerir o projeto;
- Criação de planos de *releases* para diferentes projetos usando códigos de cores, linhas cronológicas, etc;

- Criação de planos de iterações, onde se pode criar e monitorizar as *user stories* e os defeitos;
- Gestão de tarefas: atribuição, acompanhamento e controlo;
- Integração com outras ferramentas como o Visual Studio, Eclipse, etc;
- Customização de campos.

2.3.3.2 VersionOne

O VersionOne é uma ferramenta web comercial concebida para o apoio na gestão de projetos com processos ágeis. Suporta processos ágeis como: Scrum, XP, AgileUP e DSDM. Ideal para grandes organizações, uma vez que é muito completa e possui algumas características diferenciadoras, como a associação de itens do *Backlog* a objetivos estratégicos de negócio e a grupos de funcionalidades. Incorpora os conceitos relevantes dos princípios ágeis (VersionOne, 2013).

Apresenta como principais funcionalidades (VersionOne, 2013):

- Planeamento do produto, iterações, *releases*, *user stories*, tarefas e testes de aceitação;
- Visualização de quadro de *user stories*, tarefas e testes em separado;
- Acompanhamento de *bugs*;
- Geração automática de relatórios e gráficos *burndown*;
- Geração automática de histórias a partir de sugestões submetidas pelo cliente.

2.3.3.3 Pivotal Tracker

Pivotal Tracker é uma ferramenta web gratuita baseada em *user stories*. Apresenta-se como uma aplicação simples de gestão colaborativa. Suporta o processo Scrum e compreende três conceitos chaves: interface simples, que pode ser configurada pelo próprio utilizador e em grande parte baseada no *drag-and-drop*, estado do projeto e trabalho por realizar estão sempre acessíveis e o utilizador não deve estar constantemente a planear iterações (VersionOne, 2013).

Apresenta como principais funcionalidades:

- Priorização de *user stories* e criação de etiquetas para as caracterizar;
- Promove o conceito de velocidade de iteração;
- Mecanismo automático de cálculo de futuras iterações baseadas no histórico;
- Planeamento simplificado de *releases*;

- Geração de vários tipos de gráficos *burndown*;
- Importação/exportação de *user stories* por CSV;
- Geração de relatórios de progresso.

2.3.2.4 RallyDev

RallyDev é uma ferramenta comercial de gestão de projetos de desenvolvimento de software. Ajuda as equipas de projeto providenciando a gestão ao nível dos requisitos, teste, *defects* e do planeamento. Suporta as metodologias RUP, XP e Scrum (RallyDev, 2013).

Apresenta como principais funcionalidades:

- Página com informação completa sobre as histórias, com possibilidade de associar os casos de teste e *defects*;
- Definição de severidade e prioridade de um *defect* e a fase de desenvolvimento onde foi detetado;
- Personalização de *dashboard*, priorização de *Backlog* fazendo *drag-and-drop*;
- Integração com outras ferramentas.

2.3.4 Estudo Comparativo

Segundo Roldão (2000), qualquer suporte informático que uma empresa escolha para apoiar a gestão de um projeto, deve ter as seguintes capacidades mínimas:

- Criar e organizar um projeto;
- Criar recursos e custos;
- Fazer a monitorização do progresso;
- Integrar bases de dados e relatórios.

Para Conlin e Retik (1997), as empresas devem ter em consideração, na escolha do pacote de software a implementar, os seguintes aspetos:

- A interface com o utilizador deve ser acessível a utilizadores pouco experientes, mas não demasiado simplista para não afastar gestores mais experientes;
- A capacidade de monitorização do projeto é outra característica fundamental, uma vez que, deve ser possível parar o projeto e introduzir os dados reais, para assim serem comparados com o que foi planeado;

- A definição das relações de precedência entre as atividades assim como os tempos dos atrasos;
- A possibilidade de alocar recursos é uma característica muito importante nas ferramentas de gestão de projetos. Esta importância torna-se ainda mais acentuada quando o utilizador exige controlar os recursos e os custos, evitando atrasos;
- O programa pode ser capaz de lidar com custos variáveis para cada atividade, que são determinados pelas definições dos recursos e dos seus custos unitários, ou então, pode permitir só definir custos fixos para cada atividade;
- Outra característica essencial, é a capacidade de apresentar relatórios às pessoas ou gestores que podem estudar a informação, e caso seja necessário, atuarem e comunicarem essas informações aos restantes elementos envolvidos no projeto;
- A capacidade de importar e exportar dados com a finalidade de os incluir em relatórios, gráficos e/ou bases de dados é outra característica bastante útil e importante na seleção de um pacote informático.

Nesta perspetiva, foram definidos os seguintes **parâmetros de comparação**:

- **Facilidade de utilização** – *user-friendly*, a aplicação é fácil ou não de usar sem necessidade de formação/treinamento;
- **Funcionalidade de planeamento e controlo** – programação e monitorização do plano das atividades do projeto, incluindo o diagrama de rede, relações entre tarefas e estimativas, determinação do caminho crítico;
- **Gestão de recursos** – gestão de todos os recursos do projeto (pessoas, equipamentos e materiais);
- **Controlo de custos** – definição e controlo de todos os custos associados ao projeto;
- **Gestão colaborativa** – fórum de discussão, grupos, chats, notificações de email, caixa de correio interno, notas, links, lembretes e partilhas de informação;
- **Relatórios e gráficos automáticos** – visualização de relatórios e gráficos em tempo real. Pode incluir gráfico de Gantt, filtros, customização de campos, exportação para PDF e CSV;

- **Gestão de documentos** – *upload* de ficheiros e sua associação a diferentes partes do projeto, bem como o controlo de versões e níveis de acesso;
- **Multi-projeto** – capacidade para gerir vários projetos em simultâneo;
- **Uso das metodologias de GP** – gestão de projeto de acordo com uma determinada metodologia de gestão de projetos;
- **Aplicação móvel** – a aplicação é acessível através de dispositivos móveis (*smartphones* ou *tablets*) na web ou nativamente.

Ferramentas	Parâmetros de Comparação									
	A	B	C	D	E	F	G	H	I	J
Tradicionais - Desktop (Offline)										
01 MS Project 2007	2	4	4	4	N	4	N	S	0	N
02 OpenProj	4	3	2	3	N	4	N	N	0	N
03 Gantt Project	4	2	1	0	N	1	N	N	0	N
04 Open Workbench	1	3	4	2	N	2	N	S	0	N
Tradicionais – Web (Online)										
05 Basecamp	4	1	0	0	S	2	S	S	0	N
06 Redmine	3	2	-	-	S	3	S	S	0	N
07 dotProject	4	3	2	3	S	3	S	S	0	N
08 Ganttcloud	3	2	3	2	S	4	N	S	0	S
09 Do.com	3	2	1	0	S	1	S	S	0	S
10 Ace Project	3	3	3	2	S	4	S	S	0	S
Ágeis										
11 TargetProcess	1	3	3	-	S	4	S	S	2	N
12 VersionOne	1	3	-	-	S	4	S	S	3	N
13 Pivotal Tracker	2	3	-	-	S	3	S	S	2	N
14 RallyDev	1	3	-	-	S	3	S	S	3	N

Tabela 3: Comparação das ferramentas analisadas

Legenda:

- **Parâmetros:** A=Facilidade de utilização; B=Funcionalidade de planeamento e controlo; C= Gestão de recursos; D=Controlo de custos; E=Gestão colaborativa; F= Relatórios e gráficos automáticos; G= Gestão de documentos; H=Multi-projeto; I=Uso das metodologias de GP; J=Versão móvel; K=Tipo licença.
- **Comentários:** 0 = Inexistente; 1= Fraco; 2= Médio; 3=Bom; 4= Muito Bom; S= Sim; N= Não.

Tendo em conta os critérios de avaliação, no grupo das ferramentas tradicionais, destaca-se o **MS Project**, com uma forte componente de planeamento, controlo e gestão de recursos e custos, mas sem suporte a ambientes colaborativos e que exige algum

treinamento para um uso eficiente. Ainda dentro deste grupo, está a ferramenta **Gantt Project** que tal como a **MS Project** é muito conhecida. É simples e fácil de usar sem necessidade de formação, mas apresenta poucas funcionalidades. Em relação à gestão de recursos, apenas permite gerir recursos do tipo “*pessoas*”, sem possibilidade de definir custos, o que torna impossível obter o orçamento do projeto.

No grupo das ferramentas baseadas na web está a **dotProject**, que é uma ferramenta simples de usar e que apresenta todas as funcionalidades básicas necessárias para fazer uma gestão adequada de um projeto e ainda com a vantagem de ser gratuita.

Em relação às ferramentas acessíveis nos dispositivos móveis, a **AceProject** é a mais completa, com boa capacidade de apresentação de relatórios, com uma interface simples e prática.

As ferramentas de desenvolvimento ágil apresentam um défice em relação à facilidade de uso. A **Pivotal Tracker** apresenta-se como a mais simples de usar (mostrando dicas de utilização em *pop-ups* à medida que se navega), e também a menos completa, mas com a vantagem de ser gratuita. As restantes embora sejam mais completas, apresentam demasiados menus, principalmente a **TargetProcess**, o que as torna um pouco confusas e menos *user-friendly*. Estas ferramentas pelas suas particularidades irão ser analisadas seguidamente segundo outros critérios de avaliação.

2.3.4.1 Comparação entre as Ferramentas que Suportam os Processos de Desenvolvimento Ágil

Tendo em conta as premissas do processo do desenvolvimento ágil (descrito no capítulo 3), as ferramentas são comparadas a nível das seguintes áreas do conhecimento, que estão alinhadas com as definidas pelo PMBOK: Gestão de Requisitos; Gestão da Qualidade; Gestão de Tempo e Gestão de Pessoas.

PARÂMETROS	TargetProcess	Version One	Pivotal Tracker	RallyDev
Priorização dos Backlogs	Sim	Sim	Sim	Sim
Tipos de itens do Backlog	Feature, User Story e Bug	Feature, Enhancement e Defect	Feature, Bug, Chore e Release	User Story, Defect e Defect suite
Estimação dos itens	Horas	Story Points	Story Points	Story Points

Divisão de histórias em sub-histórias	Sim	Sim	Não	Sim
---------------------------------------	-----	-----	-----	-----

Tabela 4: Comparação ao nível de gestão de requisitos (baseado em (Pinto, 2010))

Ambas as ferramentas analisadas fazem a priorização dos *Backlogs*, usando a funcionalidade *drag-and-drop*. Os conceitos de história de utilização (*feature* no **VersionOne** e **Pivotal Tracker**) e *defect* (*bug* no **Pivotal Tracker** e **TargetProcess**) estão presentes. O *chore* (no **Pivotal Tracker**) é uma história que é necessária mas que não representa valor de negócio, e o *defect suite* (no **Rally Dev**) é um agrupamento de *defects*. A estimação dos itens do *Backlog* é feita maioritariamente utilizando *Story Points*. A **Pivotal Tracker** só permite a estimação dos itens do tipo *feature* (história). Com a exceção do **Pivotal Tracker**, todas as outras ferramentas fazem a divisão de histórias em sub-histórias.

PARÂMETROS	TargetProcess	VersionOne	Pivotal Tracker	RallyDev
Gestão de Testes	Integração com Selenium, Nunit e JUnit	Integração com Fitnesse e HP Quicktest Pro	Não	Integração com Fitnesse
Gestão de Defects	Integração com Bugzilla e JIRA e TestTrack Pro	Integração com Bugzilla e JIRA	Sim	Integração com Bugzilla e JIRA

Tabela 5: Comparação ao nível de gestão de qualidade (baseado em (Pinto, 2010))

Todas as ferramentas fazem integração com ferramentas de testes/*defects*, com exceção do **Pivotal Tracker** que faz a gestão de um *defect* alterando o estado do mesmo.

PARÂMETROS	TargetProcess	VersionOne	Pivotal Tracker	RallyDev
Releases	Sim	Sim	Sim	Sim
Sprints	Sim	Sim	Sim	Sim
Estimação das releases	Dias	Story Points e Dias	Dias	Story Points e Dias
Estimação das iterações	Dias Horas	Story Points e Dias	Story Points	Story Points e Dias
Estimação das tarefas	Horas	Horas	Não	Horas
Monitorização de histórias	Storyboard	Storyboard	Sim	Sim
Monitorização de tarefas	Taskboard	Taskboard	Não	Taskboard

Tabela 6: Comparação ao nível de gestão de tempo (baseado em (Pinto, 2010))

As ferramentas analisadas permitem a definição de *releases* e iterações. **VersionOne** e **RallyDev** fazem a estimação destes conceitos de forma idêntica, definindo um intervalo de dias e uma quantidade de *Story Points* que uma *release*/iteração deve perfazer. O

TargetProcess estima as iterações de forma semelhante, salvaguardando o facto de que, antes de mencionar um valor em *Story Points*, define um valor em horas que indica o esforço estimado para essa iteração. No **Pivotal Tracker** o planeamento de iterações é feito manipulando o campo *Velocity* que define a quantidade de *Story Points* que uma iteração deve perfazer. Algumas dessas ferramentas possuem uma *Storyboard* ou uma *Taskboard* de auxílio na monitorização de histórias e tarefas respetivamente.

PARÂMETROS	TargetProcess	VersionOne	Pivotal Tracker	RallyDev
Definição de equipas	Sim	Sim	Sim	Sim
Definição de papéis de um processo	Sim	Sim	Não	Sim
Indicação do trabalho alocado a cada elemento	Sim	Sim	Sim	Sim

Tabela 7: Comparação ao nível de gestão de pessoas (baseado em (Pinto, 2010))

Com exceção do **Pivotal Tracker**, todas as ferramentas permitem a definição de papéis relativos aos processos ágeis. Ambas permitem que cada elemento visualize o trabalho que tem alocado.

2.4 Conclusão

O sucesso de qualquer projeto passa pela forma como o mesmo é gerido, e este sucesso também depende da inteligência na escolha da metodologia e ferramenta de apoio a serem adotados na sua gestão. Essas escolhas podem depender da complexidade do projeto, setor de atividade e número de pessoas envolvidas. Conforme disse Roldão (2000) a ferramenta eleita deve ter certas capacidades mínimas.

Não basta só ter uma boa ferramenta de GP, uma vez que este não substitui as habilidades interpessoais de um gerente (Kerzner, 2005), por isso, é importante reforçar que uma ferramenta é apenas uma ferramenta que tem como objetivo auxiliar a GP se for bem utilizada.

3 Mapeamento do PMBOK com as Metodologias de GP

Este capítulo irá apresentar o trabalho com o mapeamento realizado entre as áreas de conhecimento bem como dos diferentes processos e as diferentes metodologias de gestão de projetos.

3.1 Áreas de Conhecimento e Processos do PMBOK

O PMBOK *Guide* apresenta cinco grupos de processos de gestão de projetos (ver Figura 1 no Capítulo 2 – Estado da Arte), resultando num total de quarenta e dois processos distribuídos por nove áreas de conhecimento, conforme se pode ver na Figura 9 em baixo. Os detalhes de cada processo podem ser consultados no **Anexo A**.



Figura 9: Processos de Gestão de Projetos. Fonte: (adaptado de PMBOK)

Um processo é um conjunto de ações e atividades inter-relacionadas, que são executadas para alcançar um produto, resultado ou serviço predefinido. Cada processo é caracterizado por **entradas, ferramentas e técnicas** aplicáveis e **saídas** resultantes. A Figura 10 exemplifica os processos da área de conhecimento de gestão de custo.

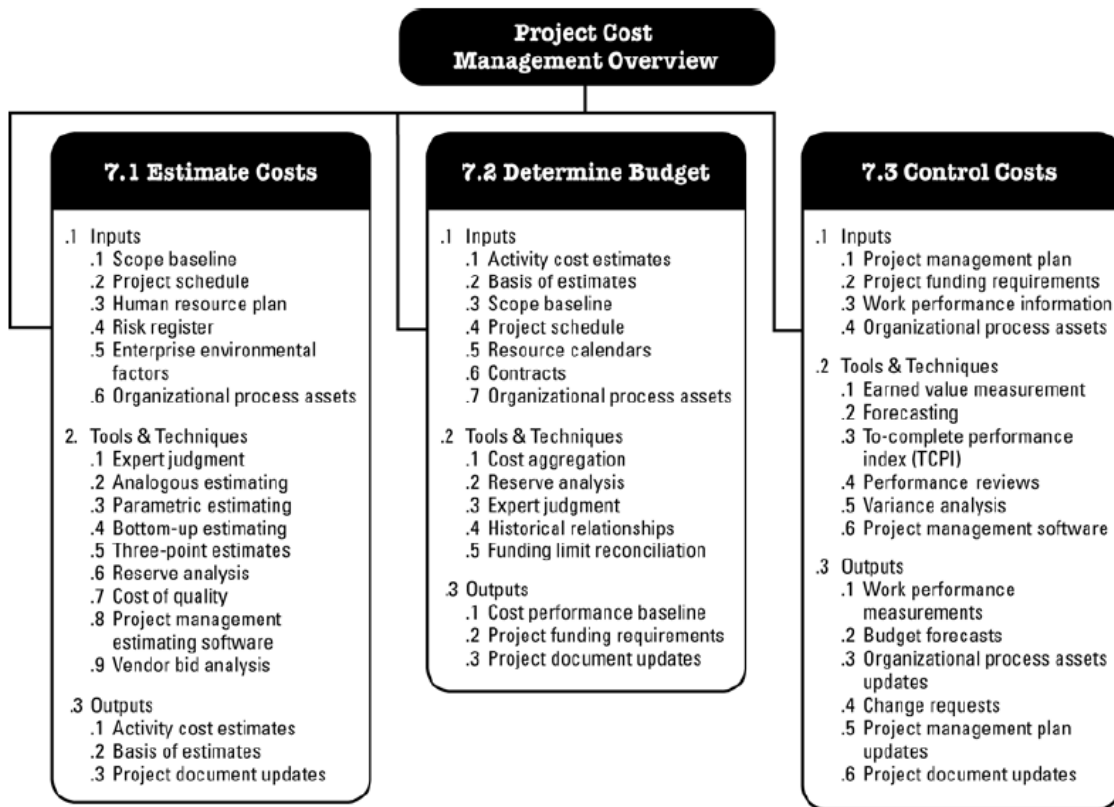


Figura 10: Resumo gestão de custos do projeto. Fonte: (PMBOK)

Os grupos de processos, conforme definidos no PMBOK, **não são fases do projeto**. Raramente os grupos de processos são eventos distintos ou que ocorrem uma única vez, são atividades sobrepostas que ocorrem ao longo de todo o projeto. A saída de um processo em geral torna uma entrada em outro processo ou é uma entrega do projeto. O grupo de processo de planejamento fornece ao grupo de processos de execução o plano de gestão e os documentos do projeto à medida que o projeto avança, com frequência envolve atualizações no plano de gestão e documentos do projeto.

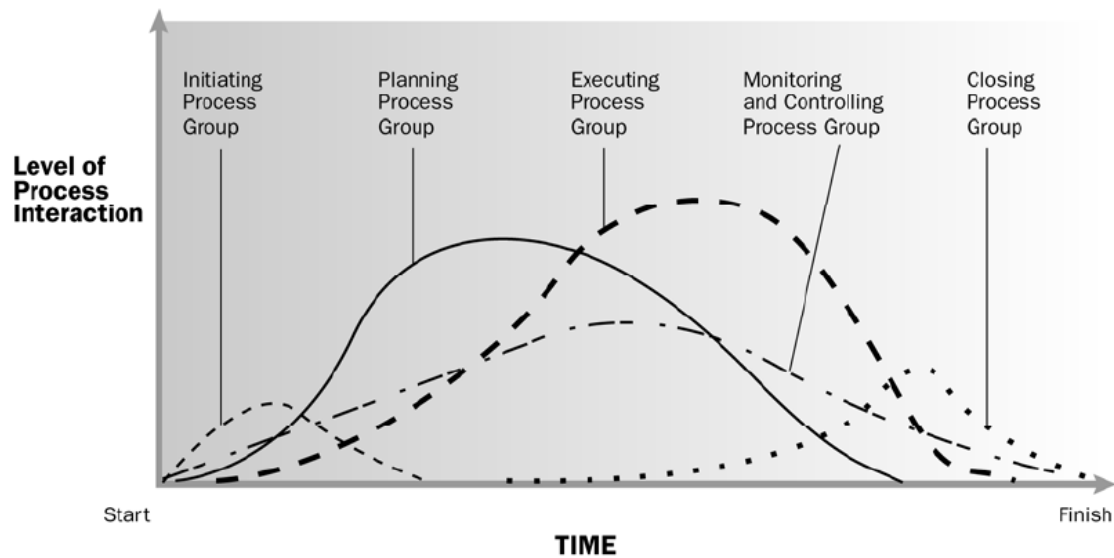


Figura 11: Interação entre grupo de processos e fases ou projeto. Fonte: (PMBOK)

A Figura 11 ilustra como os grupos de processos interagem, e mostra o nível de sobreposição em diversas ocasiões. Se o projeto estiver dividido em fases, os grupos de processos interagem dentro de cada fase.

3.2 Gestão de Projetos de SI

Um projeto de sistemas de informação tem duas dimensões principais de atuação: engenharia e gestão de projeto (Jalote, 2002). A dimensão engenharia lida com a construção do sistema e concentra-se em questões como a forma de projetar, testar código, e assim por diante. A dimensão de gestão de projeto trata adequadamente o planejamento e controla as atividades de engenharia para atender os objetivos do projeto.

Se o projeto é pequeno (por exemplo, com uma ou duas pessoas a trabalhar por algumas semanas), pode ser executado informalmente. O plano do projeto pode ser um e-mail especificando a data de entrega e talvez alguns marcos intermédios. Requisitos podem ser comunicados numa nota ou até mesmo verbalmente, e produtos de trabalhos intermédios, tais como documentos de projeto, podem ser rabiscos em blocos de anotações pessoais.

Estas técnicas informais, porém, não são aplicáveis aos projetos maiores em que muitas pessoas trabalham durante um período de tempo mais longo – o que acontece na maioria dos projetos de software comerciais. Nesses projetos, cada tarefa de engenharia deve ser feita cuidadosamente, seguindo metodologias bem testadas e os produtos do trabalho deve ser devidamente documentados para que outros possam analisá-los. As tarefas do

projeto devem ser planeadas e alocadas à equipa do projeto e controladas ao longo da execução. Em outras palavras, para executar com êxito projetos maiores, formalidades e rigor devem aumentar ao longo das duas dimensões.

Os projetos de software possuem algumas características peculiares que os distinguem dos projetos de outras áreas (Miguel, 2010):

- Implicam uma mudança contínua;
- São envolvidas pessoas de diferentes disciplinas;
- A equipa muitas vezes, trabalha em conjunto apenas num único projeto;
- A produtividade é difícil de medir;
- Os decisores estão, muitas vezes a trabalhar num domínio novo para eles;
- As linhas de autoridade não estão, muitas vezes, claramente definidas;
- Existem múltiplas visões do sucesso do projeto.

A conhecida Figura 12 em baixo, ilustra as possíveis falhas que ocorrem num projeto de software. Segundo Jalote (2002), embora haja muitas razões que levam um projeto a falhar, uma das mais importantes é a gestão inadequada.

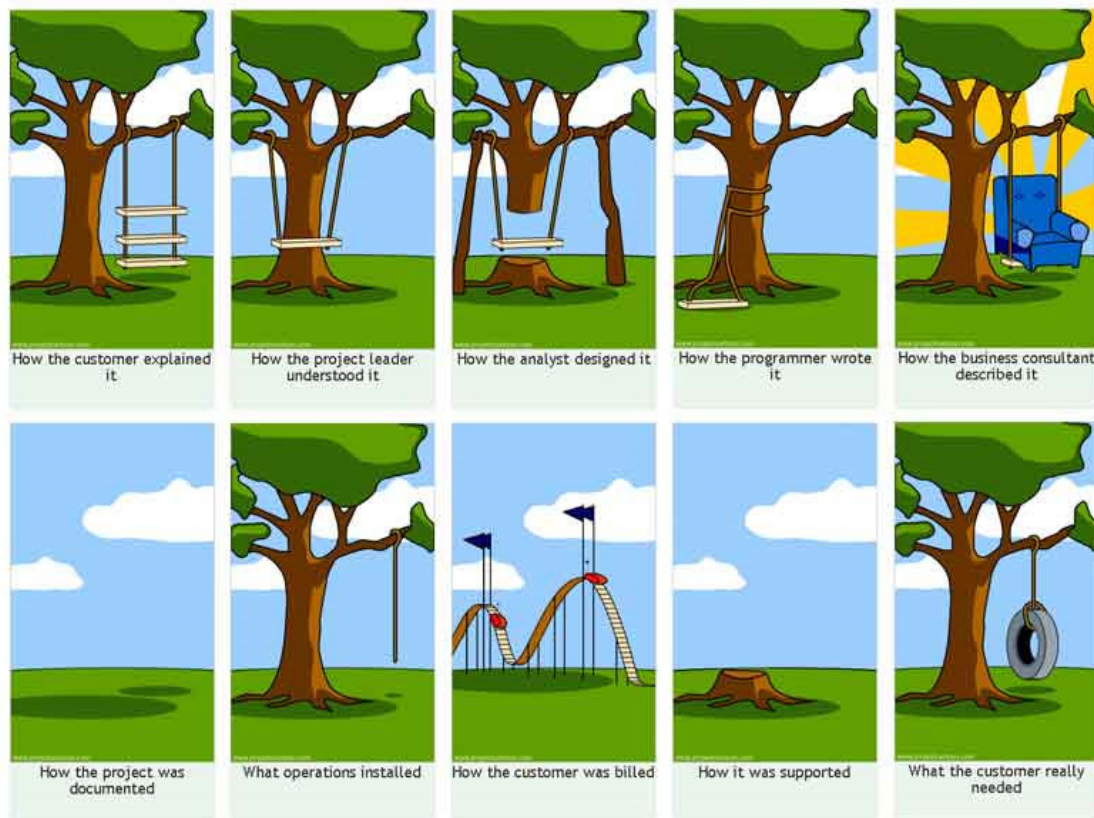


Figura 12: Ilustração das possíveis falhas no processo de desenvolvimento de software.

Para colmatar essas falhas desenvolveram-se ao longo dos anos muitas metodologias de gestão de projetos e mais recentemente surgiram as metodologias ágeis de desenvolvimento de software.

Ao contrário dos métodos tradicionais de desenvolvimento de software, como por exemplo o modelo em cascata (*waterfall*) em que existe um plano sequencial de realização das tarefas, os métodos ágeis proporcionam uma abordagem à engenharia de software que visa ajudar as equipas de desenvolvimento a responder à imprevisibilidade da construção de software através de cadências de trabalho incrementais e iterativas.

Em 2001, Kent Beck e outros⁴ 16 programadores, escritores e consultores (referido como “*Agile Alliance*”) assinaram o Manifesto para o Desenvolvimento Ágil de Software. Este manifesto define 12 princípios baseados nos valores acima referidos, para alcançar a agilidade (Pressman, 2005):

⁴ Mike Beedle; Arie van Bennekum, Alistair Cockburn; Ward Cunningham; Martin Fowler; James Grenning; Jim Highsmith; Andrew Hunt; Ron Jeffries; Jon Kern; Brian Marick; Robert C. Martin; Steve Mellor; Ken Schwaber; Jeff Sutherland e Dave Thomas.

1. Nossa maior prioridade é satisfazer o consumidor através de entregas rápidas e constantes de software funcional;
2. Mudanças de requisitos são bem-vindas, mesmo que tardias. Processos ágeis valorizam as mudanças de requisitos em prol da vantagem competitiva que fornecem aos clientes;
3. Entregar software funcional frequentemente, no intervalo de no mínimo duas semanas até o máximo de dois meses, com a preferência pelos intervalos menores;
4. Clientes e programadores devem trabalhar juntos diariamente durante todo o projeto;
5. Desenvolva projetos com pessoas motivadas. Promova o ambiente e suporte necessários, e confie que eles farão um bom trabalho;
6. O método mais eficaz e eficiente de transmitir informação para e dentro dos membros da equipa é a conversa cara a cara;
7. Software funcional é a forma básica para medir o progresso;
8. Processos ágeis promovem desenvolvimento sustentável de software. Os patrocinadores, programadores, e utilizadores deveriam manter o mesmo ritmo até ao fim;
9. Atenção contínua à excelência técnica e um bom desenho de software promovem agilidade;
10. Simplicidade - a arte de maximizar a quantidade de trabalho que não é feito - é essencial;
11. As melhores arquiteturas, requisitos, e desenhos de software surgem das equipas auto-organizadas;
12. Em intervalos regulares, a equipa reflete sobre a forma de como tornar-se mais eficaz, então altera e ajusta o seu comportamento de acordo com essa forma.

Atualmente, existem várias metodologias ágeis para desenvolvimento de software, que na verdade, são diferentes interpretações e aplicações dos princípios e valores definidos pela *Agile Alliance* em 2001. É comum, ainda, as combinações de diferentes metodologias, e também adaptações das mesmas, a fim de atender as necessidades específicas das empresas que as implementam (Schwaber e Beedle, 2002).

Seguidamente, é o feito o mapeamento tanto de metodologias traicionais de gestão de projeto como de metodologias ágeis, com o PMBOK, de modo a perceber a relação existente entre as técnicas/métodos de cada um.

As metodologias escolhidas para o mapeamento com o PMBOK são: **RUP**, **PRINCE2**, **XP** e **SCRUM** (descritas anteriormente no Capítulo 2 – Estado da Arte). Esta escolha deve-se ao facto de serem as mais conhecidas.

PMBOK	RUP	PRINCE2	XP	SCRUM
Grupo de Processos				
❖ Início	✓	✓	✓	✓
❖ Planeamento	✓	✓	✓	✓
❖ Execução	✓	✓	✓	✓
❖ Monitorização e Controlo	✓	✓	✓	✓
❖ Encerramento	✓	✓	✓	✓
Áreas de Conhecimento				
❖ 4. Gestão da Integração	✓	✓	✓	✓
❖ 5. Gestão do Âmbito	✓	✓	✓	✓
❖ 6. Gestão do Tempo	✓	✓	✓	✓
❖ 7. Gestão do Custo	✗	✓		✓
❖ 8. Gestão da Qualidade	✓	✓	✓	✓
❖ 9. Gestão de Recursos Humanos	✓	✓	✓	✓
❖ 10. Gestão da Comunicação	✓	✓	✓	✓
❖ 11. Gestão dos Riscos	✓	✓	✓	✓
❖ 12. Gestão de Aquisições	✗	✗		✓

Tabela 8: Mapeamento entre Áreas de Conhecimento e Metodologias

	Início	Planeamento	Execução	Monitorização e Controlo	Encerramento
RUP	<i>Inception</i>	<i>Elaboration</i>	<i>Construction</i>	<i>Construction</i>	<i>Transition</i>
PRINCE2	<i>Starting Up, Directing, Managing Stage Boundaries</i>	<i>Initiating, Planning, Managing Stage Boundaries</i>	<i>Controlling a stage, Managing Product Delivery, Directing</i>	<i>Controlling a stage, Managing Product Delivery, Directing</i>	<i>Closing a Project, Managing Stage Boundaries</i>

XP	<i>Release Planning, User Stories</i>	<i>Iteration Planning</i>	<i>Development</i>	<i>Project Velocity</i>	<i>Release</i>
SCRUM	<i>Sprint Planning</i>	<i>Sprint Planning</i>	<i>Sprint Execution</i>	<i>Sprint Execution</i>	<i>Sprint Review Sprint Retrospective</i>

Tabela 9: Mapeamento entre Grupo de Processos e Metodologias

Como se pode verificar na Tabela 8, todas metodologias (RUP, PRINCE2, XP e SCRUM) fazem cobertura a todos os grupos de processos do PMBOK. No RUP são cobertos com as fases, no PRINCE2 são cobertos com os respetivos processos, no XP são as práticas e no SCRUM com os eventos do mesmo, conforme mostra a Tabela 9. Ainda na Tabela 8 confere-se que o SCRUM é a única metodologia que cobre todas as áreas de conhecimento do PMBOK, com todas as áreas assinaladas a verde. As áreas com a cor vermelha significa que não existe cobertura, as de cor amarela significa que não são cobertura de uma forma completa e por fim as áreas marcadas com a cor cinzenta quer dizer que não foi possível confirmar se faz ou não a cobertura.

3.3 Mapeamento entre o RUP e o PMBOK

O RUP apresenta nove disciplinas (Figura 4). Segue uma breve descrição de cada disciplina (IBM, 2012a):

1. **Modelação de Negócio (*Business Modelling*)** - esta disciplina usa casos de uso de negócio (*business use case*) para documentar os processos de negócio, de modo a garantir um bom entendimento entre as partes interessadas de que o processo de negócio deve ser apoiado na organização. Os casos de uso são analisados para entender como o negócio deve apoiar os processos de negócio;
2. **Requisitos (*Requirements*)** – o objetivo desta disciplina é descrever o que o sistema deve fazer e permite que os programadores e o cliente estejam em sintonia;
3. **Análise e Design (*Analysis e Design*)** – mostram como é que o sistema deve ser realizado aquando da sua implementação. Tem como resultado um modelo de desenho e opcionalmente um modelo de análise;
4. **Implementação (*Implementation*)** – tem como objetivo definir a organização do código em termos de implementação organizados em camadas, implementação de

classes e objetos como componentes, testar os componentes e integrar os resultados individuais ou em equipa;

5. **Teste (*Test*)** - nesta disciplina é efetuado a verificação da integração de todos os componentes, verificação dos requisitos e identificar as possíveis falhas;
6. **Desenvolvimento (*Deployment*)** – produzir o produto final com sucesso e entregar ao cliente. Engloba atividades como: produção de versões externas do software, empacotamento, distribuição, instalação,...;
7. **Configuração e Gestão de Mudança (*Configuration e Change Managment*)** - focam no desenvolvimento iterativo do processo. Facilita as tarefas do projetoc com o fornecimento de *framework* para a gestão de projeto, diretrizes de planeamento execução e monitorização, e ainda *framework* para gerir os riscos;
8. **Gestão de Projeto (*Project Managment*)** – descreve como controlar os vários artefactos produzidos por vários elementos da equipa do projeto. Assegura que os artefactos resultantes não entram em conflito;
9. **Ambiente (*Environment*)** – fornecer à organização de desenvolvimento de software, o ambiente de desenvolvimento ao nível de processos e ferramentas necessárias para apoiar os elementos da equipa.

A tabela seguinte mostra como o RUP se mapeia nas áreas de conhecimento do PMBOK.

PMBOK - Áreas de Conhecimento	RUP - Disciplinas
4. Gestão da Integração	Modelação de negócio, Requisitos, Gestão de projeto, Configuração e Gestão de mudanças
5. Gestão do Âmbito	Gestão de projeto; Requisitos; Configuração e Gestão de mudanças
6. Gestão do Tempo	Gestão de projeto
7. Gestão do Custo	Sem mapeamento
8. Gestão da Qualidade	Configuração e Gestão de mudança
9. Gestão de Recursos Humanos	Sem mapeamento completo, embora defina a organização do projeto
10. Gestão da Comunicação	Gestão de projeto
11. Gestão dos Riscos	Gestão de projeto
12. Gestão de Aquisições	Sem mapeamento

Tabela 10: Mapeamento entre o RUP e o PMBOK

Como se por ver na tabela anterior (Tabela 10), a maioria das áreas do PMBOK são cobertas pelas disciplinas do RUP, apenas as áreas de gestão de custo e de gestão de aquisições não são cobertas, embora a área de gestão de recursos humanos não é

completamente coberta. O RUP garante a cobertura destas áreas através de tarefas enquadradas nas disciplinas de Modelação de Negócio, Requisitos, Configuração e Gestão de Mudança e Gestão de Projeto (English, 2013).

- **Gestão da Integração** – garantida através das principais tarefas como: desenvolvimento do modelo de negócio, planeamento de fases e iterações, programação e atribuição de trabalho, planeamento e revisão do plano de aceitação, controlo do estado do projeto e preparação do fecho do projeto;
- **Gestão do Âmbito** - revisão do ciclo de vida e *milestone* do projeto e planeamento e avaliação da iteração;
- **Gestão do Tempo** – desenvolvimento do plano de iteração e planeamento de fases e iterações;
- **Gestão da Qualidade** – desenvolvimento do plano de garantia de qualidade;
- **Gestão de Recursos Humanos** – definição da organização de projetos e pessoas, aquisição de pessoas, programação e atribuição de tarefas, controlo de exceções e problemas;
- **Gestão da Comunicação** – compilação do plano de desenvolvimento de software, planeamento de fases e iterações e relatório de estado;
- **Gestão dos Riscos** - desenvolvimento do plano de gestão de riscos, identificação e avaliação dos riscos.

Com base nesse mapeamento não existe nenhuma incompatibilidade fundamental entre PMBOK e RUP. Termos diferentes são usados para descrever semanticamente conceitos semelhantes ou idênticos, mas as boas práticas do PMBOK não contrariam o RUP e nem o RUP contradiz as boas práticas do PMBOK (Charbonneau, 2004).

O mapeamento mais detalhado entre o RUP e o PMBOK pode ser consultado no **Anexo B**.

3.4 Mapeamento entre o PRINCE2 e o PMBOK

O PRINCE2 para além dos processos (Figura 6) apresenta sete temas ou componentes e também sete princípios. Os temas são abordados continuamente ao longo do ciclo de vida do projeto e fornecem orientações sobre como o processo deve ser realizado (Murray, 2011). Esses temas são:

1. *Business Case*;

2. Organização;
3. Qualidade;
4. Plano;
5. Risco;
6. Mudança;
7. Progresso.

Os temas **Business Case**, **Qualidade** e **Plano** descrevem como a *baseline* para os benefícios, riscos, âmbito, qualidade, custo e tempo são estabelecidos; os temas **Qualidade**, **Risco**, **Mudança** e **Progresso** descrevem como a equipa de gestão de projeto monitoriza e controla o trabalho conforme o progresso do projeto e o tema **Organização** apoia os outros temas com uma estrutura para o projeto, definindo papéis e responsabilidades entre as equipas do projeto (Murray, 2011).

Os sete princípios do PRINCE2 são (Murray, 2011):

1. **Justificação do negócio** – o projeto no ambiente PRINCE2 deve ter uma justificação continua para o negócio;
2. **Lições tiradas da experiencia** – equipas do projeto PRINCE2 aprendem com experiências anteriores (lições são gravadas e postas em prática ao longo da vida do projeto);
3. **Papéis e responsabilidades definidos** – os papéis e responsabilidades são definidos e acordados com a organização e envolve todos os *stackholders*;
4. **Gestão por sequência** – planeamento, controlo e monitorização realizados etapa por etapa;
5. **Gestão por exceção** – definição de tolerâncias para cada objetivo do projeto visando estabelecer limites e delegar autoridades;
6. **Focalização no produto** – foca na definição e entrega de produtos;
7. **Adaptação ao contexto do projeto** – pode ser customizado para qualquer tipo de projeto.

A tabela seguinte (Tabela 11) mostra como PRINCE2 pode ser mapeado com as áreas de conhecimento do PMBOK.

PMBOK – Áreas de Conhecimento	PRINCE2 – Temas/Componentes
4. Gestão da Integração	Controlo da mudança
5. Gestão do Âmbito	Planos e <i>Business Case</i>

6. Gestão do Tempo	Planos e <i>Business Case</i>
7. Gestão do Custo	Planos e <i>Business Case</i>
8. Gestão da Qualidade	Qualidade
9. Gestão de Recursos Humanos	Organização
10. Gestão da Comunicação	Mudanças
11. Gestão dos Riscos	Riscos
12. Gestão de Aquisições	Sem mapeamento

Tabela 11: Mapeamento entre o PRINCE2 e o PMBOK

Com a exceção da área de gestão de aquisições, é possível fazer um mapeamento entre as áreas de conhecimento do PMBOK e os diferentes temas/componentes do PRINCE2 (Coronado, 2008).

- **Gestão da Integração** – análise e avaliação do impacto da mudança;
- **Gestão do Âmbito, Tempo e Custo** – definir mecanismos de avaliação da viabilidade do projeto como meio de ajudar na tomada de decisão, análise do custo benefício e desenvolvimento do plano do projeto definindo atividades recursos e tempo;
- **Gestão da Qualidade** – definir o método de verificação do produto de modo a garantir que satisfaz os requisitos desejados;
- **Gestão de Recursos Humanos** – esta área é limitada no PRINCE2, mas a organização desenvolve a estrutura de projeto, definindo papéis e responsabilidades;
- **Gestão da Comunicação** – comunicação de mudanças durante todo o ciclo de vida do projeto;
- **Gestão dos Riscos** – controle e avaliação dos riscos, planejamento e respostas aos mesmos.

O PMBOK e o PRINCE2 são compatíveis e complementares, não são incapazes de coexistir e não competem entre si. Tal como o PMBOK, o PRINCE2 também pode ser aplicado a qualquer tipo de projeto.

3.5 Mapeamento entre o Extreme Programming (XP) e o PMBOK

De acordo com Beck (1999), XP é uma forma de promover a excelência no desenvolvimento de software, e distingue-se de outras técnicas pelas seguintes características:

- Trabalha com iterações de duração bastante curta, que resulta em respostas rápidas, concretas e contínuas;

- Usa uma abordagem incremental de planejamento, que resulta em planos abrangentes que podem evoluir durante o ciclo de vida do projeto;
- Tem a possibilidade de planejar melhor a implementação das funcionalidades do software, permitindo responder mais rápido às necessidades de mudanças
- Usa mecanismos automatizados para realizar testes, que podem ser desenvolvidos pelos programadores, clientes ou pessoal de qualidade para monitorizar o progresso do desenvolvimento, permitindo que o sistema evolua e que os defeitos sejam identificados o mais cedo possível;
- Confia na comunicação verbal, nos testes e nos programas fontes para comunicar a estrutura do sistema;
- Trabalha com um processo evolutivo de desenho que persiste durante a vida do projeto;
- Confia na colaboração entre indivíduos ativamente contratados e com talentos específicos;
- Usa práticas que atendem às necessidades imediatas dos membros da equipa e de longo prazo do projeto.

O XP apresenta um conjunto de boas práticas que foram anteriormente descritas no Capítulo 2. A tabela seguinte (Tabela 12) mostra como XP pode ser mapeado com as áreas de conhecimento do PMBOK.

PMBOK – Áreas de Conhecimento	XP – Práticas/Valores
4. Gestão da Integração	Pequenas versões (Limitado)
5. Gestão do Âmbito	<i>User Stories</i>
6. Gestão do Tempo	<i>Planning Game</i>
7. Gestão do Custo	Sem referência
8. Gestão da Qualidade	Desenvolvimento orientado a testes
9. Gestão de Recursos Humanos	Equipa coesa
10. Gestão da Comunicação	Programação em pares; Reuniões em pé; Comunicação, <i>Feedback</i>
11. Gestão dos Riscos	<i>Feedback</i> , Testes de aceitação e Entrega de pequenas versões
12. Gestão de Aquisições	Sem referência

Tabela 12: Mapeamento entre o XP e o PMBOK

As áreas de conhecimento do PMBOK são mapeadas com as práticas/valores do XP:

- **Gestão da Integração** – XP não faz integração da equipa de desenvolvimento com outras partes. Não descreve explicitamente como integrar o trabalho das outras partes embora as práticas não excluam essa possibilidade. Pequenas entregas fazem a gestão da integração num processo contínuo em vez de documentação e testes no fim do calendário (Goebel, 2003);
- **Gestão do Âmbito** – baseia-se na escrita de *user stories*, onde é descrito todos os cenários de utilização, ou seja, o que o software deve fazer após sua conclusão. As *user stories* são escritas pelo próprio cliente/utilizador;
- **Gestão do Tempo** – no XP os prazos é definida semanalmente através do *Planning Game*. A equipa de desenvolvimento reúne com o cliente onde são priorizadas as funcionalidades;
- **Gestão do Custo** – não foi encontrada nenhuma referência para gestão de custo;
- **Gestão da Qualidade** – no desenvolvimento orientado a teste, são realizados testes unitários que são essenciais para garantir a qualidade;
- **Gestão de Recursos Humanos** – a gestão da contratação leva a ter em consideração outras habilidades para além das técnicas. Equipa é formada de forma multidisciplinar;
- **Gestão da Comunicação** – a comunicação está muito presente no XP, envolvendo todas as partes interessadas;
- **Gestão dos Riscos** – os testes de aceitação e entregas muito cedo de funcionalidades permite obter feedback do cliente e controlo dos riscos;
- **Gestão de Aquisições** - não foi encontrada nenhuma referência para gestão de aquisições.

O XP cobre quase todas as áreas de conhecimento, sendo que não foi possível confirmar a área de gestão de custo e a área de gestão de aquisições.

3.6 Mapeamento entre o SCRUM e o PMBOK

Conforme o que foi referido no Capítulo 2 (Estado da Arte) o SCRUM possui um conjunto de práticas e papéis pré-definidos:

Papéis:

- *Product Owner*
- *Scrum Master*
- *Team*

Eventos:

- *Sprint Planning*
- *Daily Scrum Meeting*
- *Sprint Retrospective*

Artefactos:

- *Product Backlog;*
- *Sprint Backlog;*
- *Burndown Charts.*

A seguir são listadas um conjunto de práticas e princípios do SCRUM (Miguel, 2010):

- Os clientes tornam-se parte integrante da equipa de desenvolvimento;
- À semelhança de todas as outras formas de desenvolvimento ágil de software, tem frequentes entregas intermédias com funcionalidade operacional. Isto permite ao cliente utilizar o software mais cedo e possibilita ao projeto mudar os requisitos à medida que as necessidades se vão alterando;
- São desenvolvidos frequentes planos de risco e de mitigação pela própria equipa de desenvolvimento. A análise e gestão dos riscos ocorrem em todos os estágios e com compromissos claros dos donos;
- Existe transparência no planeamento e no desenvolvimento dos módulos, de modo que todos sabem quem é responsável por o “quê” e “quando”;
- Há frequentes reuniões de *stackholders* para monitorizar o progresso, com atualizações de indicadores gráficos;
- Deve existir um sistema de aviso prematuro, isto é, um sistema que dê visibilidade a potenciais derrapagens ou desvios, antes destes acontecerem;
- Nenhum problema deve ser escondido. Ninguém é penalizado por reconhecer ou descrever um problema imprevisto;
- Os locais de trabalho e os períodos de trabalho devem ser organizados – “trabalhar mais horas” não significa necessariamente “realizar mais trabalho”.

A tabela seguinte (Tabela 13) mostra como SCRUM pode ser mapeado com as áreas de conhecimento do PMBOK.

PMBOK – Áreas de Conhecimento	SCRUM
4. Gestão da Integração	Equipa Scrum e <i>Product Backlog</i>

5. Gestão do Âmbito	<i>Product Backlog</i>
6. Gestão do Tempo	Equipa Scrum e <i>Sprint Planning</i>
7. Gestão do Custo	Equipa Scrum e <i>Product Owner</i>
8. Gestão da Qualidade	Equipa/Práticas Scrum e <i>Burndown Charts</i>
9. Gestão de Recursos Humanos	Equipa Scrum e Valores do Scrum
10. Gestão da Comunicação	<i>Product Owner e Burndown Charts</i>
11. Gestão dos Riscos	Equipa Scrum, Plano do <i>Sprint</i> e <i>Daily Scrum</i>
12. Gestão de Aquisições	Equipa Scrum

Tabela 13: Mapeamento entre o SCRUM e o PMBOK

As áreas de conhecimento do PMBOK são mapeadas com os processos/práticas do SCRUM (Sutherland e Ahmad, 2011):

- **Gestão da Integração** - assegurado por *Product Owner* e Equipa SCRUM. Planos de entrega; revisão do *Sprint*; gestão dos princípios do SCRUM e controlo de mudanças através do *Product Backlog*;
- **Gestão do Âmbito** - gestão de âmbito é inerentemente incorporada no processo SCRUM. O SCRUM mantém o tempo e os custos fixos, o único item negociável é o âmbito, que é fixado no início do *Sprint*;
- **Gestão do Tempo** - SCRUM utiliza uma abordagem *Top-down* para a gestão de tempo – plano de entrega, Sprints, individuais e as atividades / tarefas diárias; escolhe os recursos de valor mais elevado em vez de executar tarefas definidas no plano de projeto;
- **Gestão do Custo** - as estimativas são feitas numa abordagem *Top-down* e são revistas ao longo do ciclo de vida do projeto, controlo de custo é uma função da equipa com o *Product Owner*;
- **Gestão da Qualidade** - não há planeamento formal da qualidade em Scrum, qualidade está embutido na estrutura do SCRUM, devido à natureza dos processos, práticas e da equipa SCRUM;
- **Gestões de Recursos Humanos** - equipas SCRUM são dedicadas, multifuncional e auto-organizado, com responsabilidade mútua que exige um modelo de liderança servo para SCRUM *Masters*;
- **Gestão da Comunicação** - esta é a maior diferença entre projetos tradicionais e SCRUM, já que a natureza multifuncional de uma equipa Scrum cria e enfatiza a comunicação direta/cara a cara e frequente entre os membros da equipa e *stakeholders*, eliminando a necessidade de relatórios tradicionais; menos formalidade de um projeto SCRUM resulta numa melhor comunicação;

- **Gestão dos Riscos** - SCRUM é um sistema de redução de riscos, que lida com o risco ao nível estratégico (resposta rápida a mudanças) e nível tático (revisão da iteração, e do plano de entrega, análise SWOT, listas de verificação e *Brainstorming*). Equipa inteira está envolvida no planeamento de risco, mitigação e resposta;
- **Gestão de Aquisições** – Equipa SCRUM fornece *inputs* para a descrição das necessidades usando iterações ou prova de conceito. Vários tipos de contratos são usados em projetos SCRUM para fornecer flexibilidade para os clientes.

O SCRUM cobre todas as áreas de conhecimento do PMBOK. Sendo SCRUM uma metodologia ágil, tem uma abordagem diferente do método tradicional com é o caso do PMBOK, mas não existe conflito entre ambos, até porque SCRUM pode perfeitamente ser utilizado em conjunto com outras práticas/metodologias.

Embora o SCRUM tenha como objetivo a gestão de desenvolvimento ágil de software, pode igualmente ser utilizado para operação de equipas de manutenção (Miguel, 2010).

O mapeamento mais detalhado entre o SCRUM e o PMBOK pode ser consultado no **Anexo C**.

3.7 Conclusão

Como se pode observar e comprovar nas tabelas de mapeamento entre PMBOK e as metodologias referenciadas (Tabela 8 e Tabela 9), todas elas fazem mapeamento com os grupos de processos e com quase todas as áreas de conhecimento, com a maior penalização a encontrar-se na área de gestão das aquisições.

Destaca-se como principal similaridade entre PMBOK e as metodologias abordadas, o facto de ambos se preocuparem em entregar o produto dentro do prazo, no tempo planeado e com qualidade, embora cada um possua a sua abordagem de como chegar a esse objetivo. Fases e subfases existem em ambos, a maneira como são interpretadas é que é diferente.

Como diferença destaca-se o facto de o PMBOK ser um *standard* genérico que pode ser aplicado a qualquer tipo de projeto, e as metodologias serem essencialmente orientadas para projetos de software. O PMBOK tem uma abordagem descritiva (explica as

técnicas de gestão de projeto) enquanto que, o RUP e o PRINCE2 têm uma abordagem prescritiva (explicam como técnicas de gestão de projetos devem ser estruturadas e aplicadas) e as metodologias ágeis XP e SCRUM têm uma abordagem empírica (uso do resultado do projeto no controlo do mesmo, em vez de um plano prescrito).

O PMBOK está orientado aos processos, mas isto não quer dizer que todos os processos devem ser aplicados em todos os projetos. Não existem divergências entre o PMBOK e as metodologias uma vez que ambos acabam por se complementa

4 Processo de Implementação da Aplicação

A aplicação desenvolvida é uma aplicação de apoio a gestão de projetos de desenvolvimento de SI. A mesma está disponível através da web e em dispositivos móveis.

Através desta aplicação é possível ter acesso ao mapeamento das metodologias com os grupos de processos e áreas de conhecimento definidos no PMBOK, conforme apresentado no capítulo anterior. Para além dessas informações é possível ter acesso aos seguintes dados:

- Todas as áreas de conhecimento e grupos de processos;
- Mapeamento entre áreas de conhecimento e grupos de processos;
- Todas as metodologias de gestão de projetos descritas no capítulo 2;
- Todas as ferramentas de gestão de projetos descritas no capítulo 2;
- As disciplinas e fases do RUP;
- Os processos e os temas/componentes do PRINCE2;
- As boas práticas/valores do XP;
- Os papéis, eventos, artefactos, princípios e práticas do SCRUM;
- Também é possível ver um glossário com os principais termos relacionados com a gestão de projetos.

Para ter acesso a estas informações é preciso efetuar um registo prévio e posterior autenticação. Os utilizadores para além de terem acesso às informações acima referidas, podem visualizar e alterar os seus dados de perfil e contactar o administrador do sistema, através do preenchimento de um formulário.

O administrador do sistema tem acesso ao histórico de registo e autenticação dos utilizadores. Tem uma área de gestão de toda a informação, onde pode inserir, alterar e apagar dados.

4.1 Arquitetura do Sistema

A aplicação foi desenvolvida utilizando o **Sencha Touch** (versão 2.1.1). **Sencha Touch** é uma *framework* baseada em HTML5, CSS3 e JavaScript que permite desenvolver aplicações web para dispositivos móveis, destacando o facto do código produzido

funcionar em várias plataformas/ambientes – “*Write Once, Run Everywhere*”, ou seja, a aplicação pode funcionar em plataformas iOS, Android, BlackBerry, entre outros, utilizando o mesmo código. Por outro lado, esta aplicação funciona igualmente bem nos *browsers* Google Chrome e Safari num computador desktop. Entre outras características estão o componente de UI (User Interface), configuração do componente de acordo com o dispositivo alvo e a capacidade de aceder aos serviços do dispositivo como por exemplo a câmara, caso utilizado com outro *framework* (Sencha Touch, 2013).

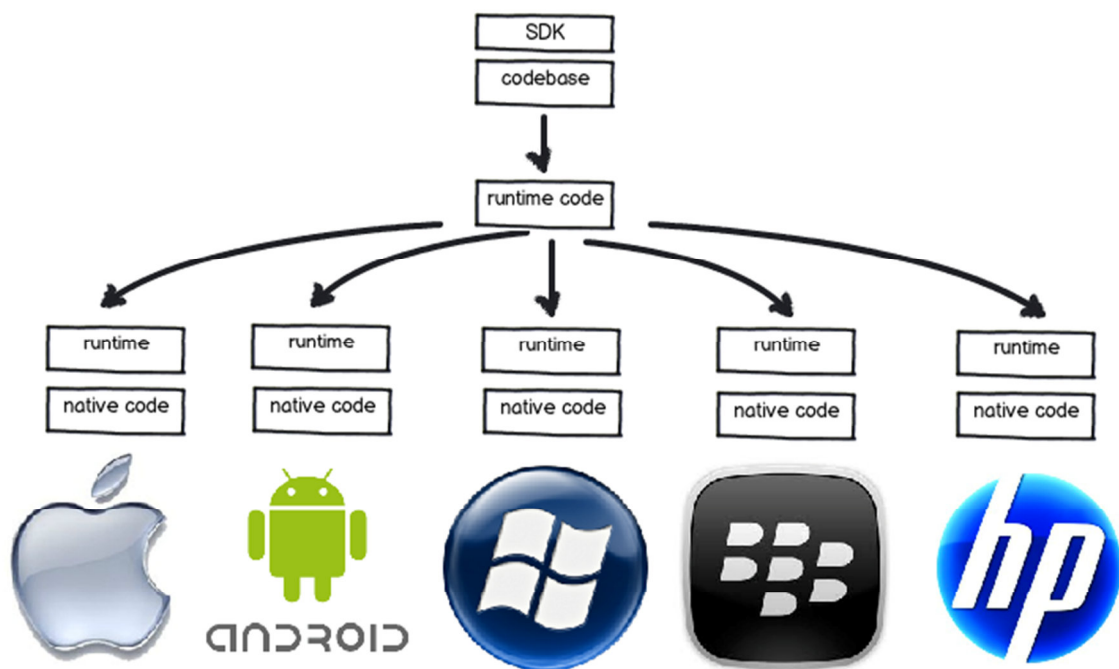


Figura 13: “Write Once, Run Everywhere”. Fonte: (Sencha Touch, 2013)

A *framework* de desenvolvimento oferece igualmente uma arquitetura baseada em MVC – Model View Controller (Figura 14).

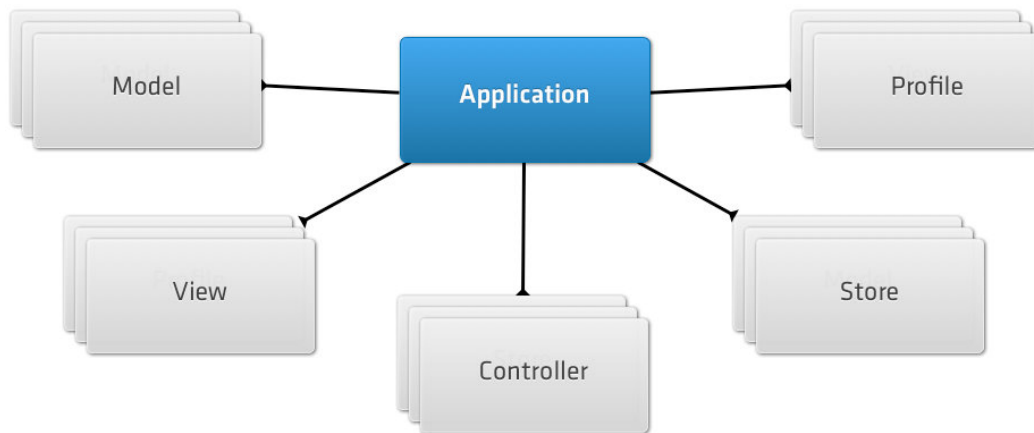


Figura 14: Arquitetura MVC do Sencha Touch. Fonte: (Sencha Touch, 2013)

- **Model** - representa o tipo de dados que devem ser usados/armazenados na aplicação;
- **View** - responsável por mostrar os dados ao utilizador;
- **Controller** – lida com a interação do utilizador e interface e a interação entre View e Model;
- **Store** – responsável por carregar os dados para a aplicação;
- **Profile** – ajuda na personalização da interface do utilizador para vários tipos de dispositivos.

A imagem em baixo mostra a estrutura genérica de um projeto em Sencha Touch.

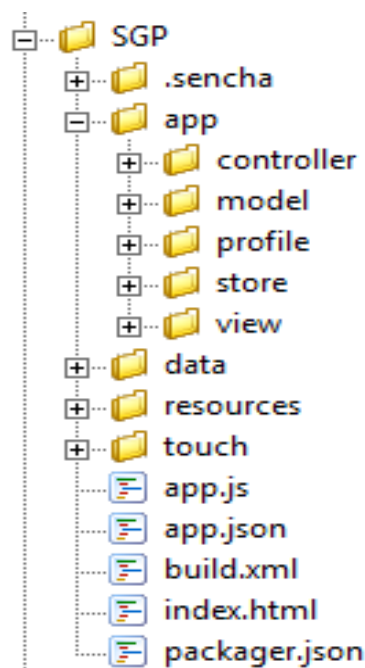


Figura 15: Estrutura do projeto em Sencha Touch

1. Index.html – página onde a aplicação será alojada;
2. Diretoria **app** – onde estão as subdiretorias Model, View, Controller, Store e Profile;
3. Diretoria **resources** – contém imagens, css;
4. Diretoria **touch** – contém os ficheiros do *framework*;
5. App.js – contém configurações globais do sistema, referência para todos os ficheiros do Model, View, Controller, Store e Profile.

A diretoria **data** não tem que necessariamente fazer parte do projeto, neste caso foi necessário uma vez que os dados em vez de serem armazenados no *localStorage* que é utilizado pelo **Sencha Touch** para guardar os dados localmente no *browser* do cliente, foram armazenados numa base de dados MySQL. Nesta diretoria estão todos os ficheiros de comunicação com o servidor. A linguagem utilizada foi o PHP, a comunicação foi feita utilizando o AJAX e os dados são devolvidos no formato JSON.

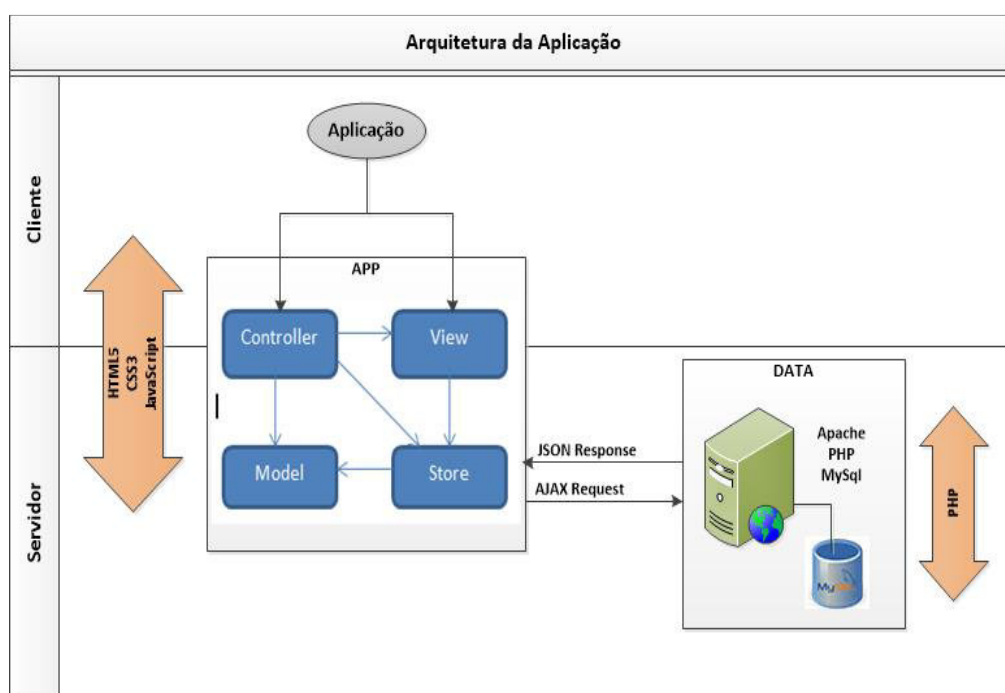


Figura 16: Arquitetura da aplicação desenvolvida

A figura anterior (Figura 16) representa a arquitetura da aplicação. Conforme já referido, foi utilizado a *framework* Sencha Touch, com os dados a serem guardados num servidor web. De seguida apresenta-se um exemplo de código que demonstra como é

efetuada a gestão de utilizadores. É possível verificar como a aplicação comunica com o servidor usando o AJAX para carregar os dados dos utilizadores para a Store e como o servidor devolve os dados no formato JSON.

App/store/users.js

```
Ext.define('SGP.store.Users', {
    extend: 'Ext.data.Store',
    config: {
        model: 'SGP.model.User',
        autoLoad: true,
        proxy: {
            type: 'ajax',
            api: {
                read: 'data/read_user.php',
                update: 'data/update_user.php',
                create: 'data/create_user.php',
                destroy: 'data/delete_user.php'
            },
            reader: {
                type: 'json',
                rootProperty: 'data',
                successProperty: 'success'
            },
            writer: {
                type: 'json',
                rootProperty: 'data',
                encode: true
            }
        },
    },
});
```

Data/lib/response.php

```
class Response {
    public $success, $data, $total, $message, $errors, $tid, $trace;
    public function __construct($params = array()) {
        $this->success = isset($params["success"]) ?
$params["success"] : false;
        $this->message = isset($params["message"]) ?
$params["message"] : '';
        $this->total = isset($params["total"]) ?
$params["total"] : '';
        $this->data = isset($params["data"]) ?
$params["data"] : array();
    }
    public function to_json() {
        return json_encode(array(
            'success' => $this->success,
            'message' => $this->message,
            'total' => $this->total,
            'data' => $this->data
        ));
    }
}
```

Data/read_user.php

```
...
//Creating Json
$res = new Response();
```

```

$res->success = true;
$res->message = "Loaded data";
$res->total = $total;
$res->data = $array_users; // array com todos os dados do
utilizador
print_r($res->to_json());

```

4.2 Requisitos Funcionais da Aplicação

A análise e o mapeamento feito entre as metodologias de GP e o PMBOK ajudaram na definição de alguns requisitos da aplicação que vai permitir aos utilizadores terem acesso de toda a informação relacionada com as metodologias e o PMBOK de uma forma integrada.

A tabela seguinte (Tabela 14) lista os principais requisitos funcionais da aplicação.

ID	Requisitos
#R01	A aplicação deve poder ser executada num <i>browser web</i> .
#R02	A aplicação deve estar acessível através de um dispositivo móvel.
#R03	A aplicação deve permitir o registo e autenticação de utilizadores.
#R04	Os utilizadores devem poder consultar e alterar os seus dados de perfil através da aplicação.
#R05	Deve ser possível consultar na aplicação as áreas de conhecimento definidas no PMBOK.
#R06	Deve ser possível consultar na aplicação os grupos de processos definidos no PMBOK.
#R07	A aplicação deve mostrar o mapeamento entre áreas de conhecimento e grupos de processos definidos no PMBOK.
#R08	Deve ser possível consultar as metodologias de gestão de projetos de sistemas de informação.
#R10	Deve ser possível visualizar o mapeamento da metodologia de gestão de projeto com as áreas de conhecimento do PMBOK.
#R11	Deve ser possível visualizar o mapeamento da metodologia de gestão de projeto com o grupo de processos do PMBOK.
#R12	Deve ser possível consultar as principais ferramentas de gestão de projetos.
#R13	Deve ser possível o administrador consultar o histórico de registo e autenticação.
#R14	Deve ser possível o administrador consultar o número de utilizadores registados no sistema.

#R15	O administrador deve ter uma área de gestão de dados onde: insere, altera e apaga dados.
------	--

Tabela 14: Requisitos funcionais da aplicação

4.3 Protótipo da Aplicação

A aplicação foi desenhada tendo em conta as características das aplicações móveis e os requisitos descritos anteriormente. Seguidamente é explicado o funcionamento da mesma.

A aplicação tem um mecanismo de gestão de acesso com privilégios distintos (Figura 17 e Figura 18).

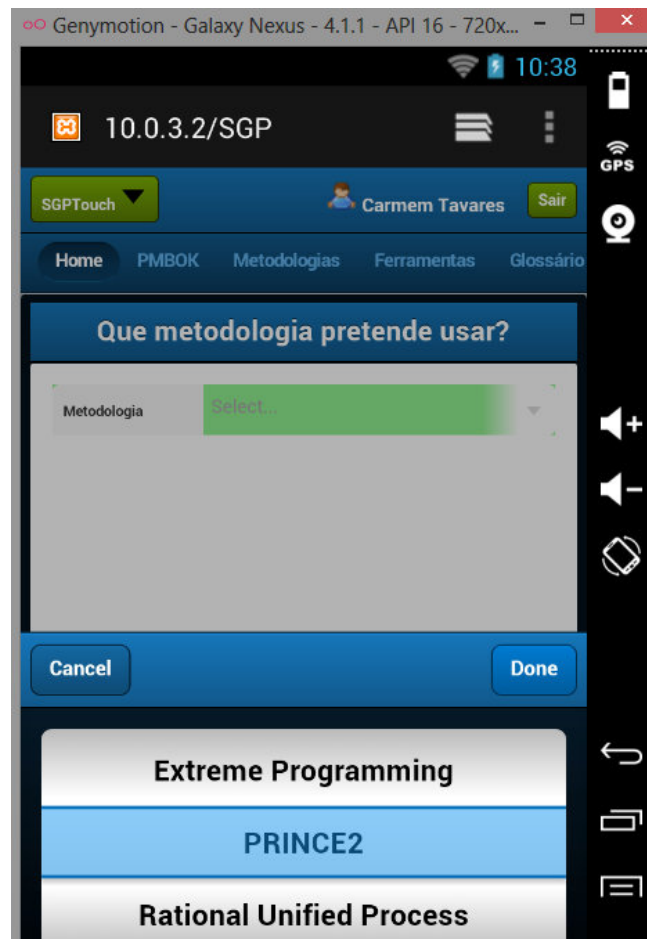


Figura 17: Ecrã inicial do utilizador

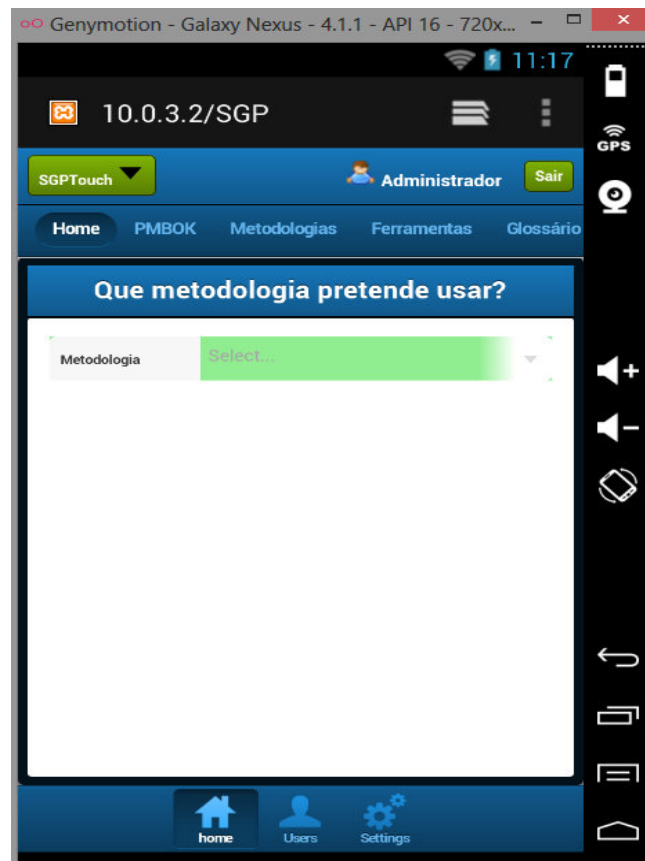


Figura 18: Ecrã inicial de administrador

Após autenticação na aplicação o utilizador terá acesso a um conjunto de informações. Na página inicial (*Home*) o utilizador pode escolher uma metodologia e ser-lhe-á apresentado o mapeamento com o PMBOK. Mais informações estão agrupadas nas seguintes opções:

- **PMBOK** – nesta opção estão disponíveis os grupos de processos e as áreas de conhecimento, que por sua vez mostram os processos correspondentes;
- **Metodologias** – nesta opção o utilizador terá acesso a uma lista de metodologias e para cada uma delas pode visualizar o mapeamento com o PMBOK e as suas respetivas características (disciplinas, processos, valores, práticas, componentes...), ver opções na Figura 19;
- **Ferramentas** – esta opção mostra a lista de ferramentas de GP organizadas por categoria;
- **Glossário** – a opção glossário mostra uma lista com os termos relacionados com a gestão de projetos e suas definições.

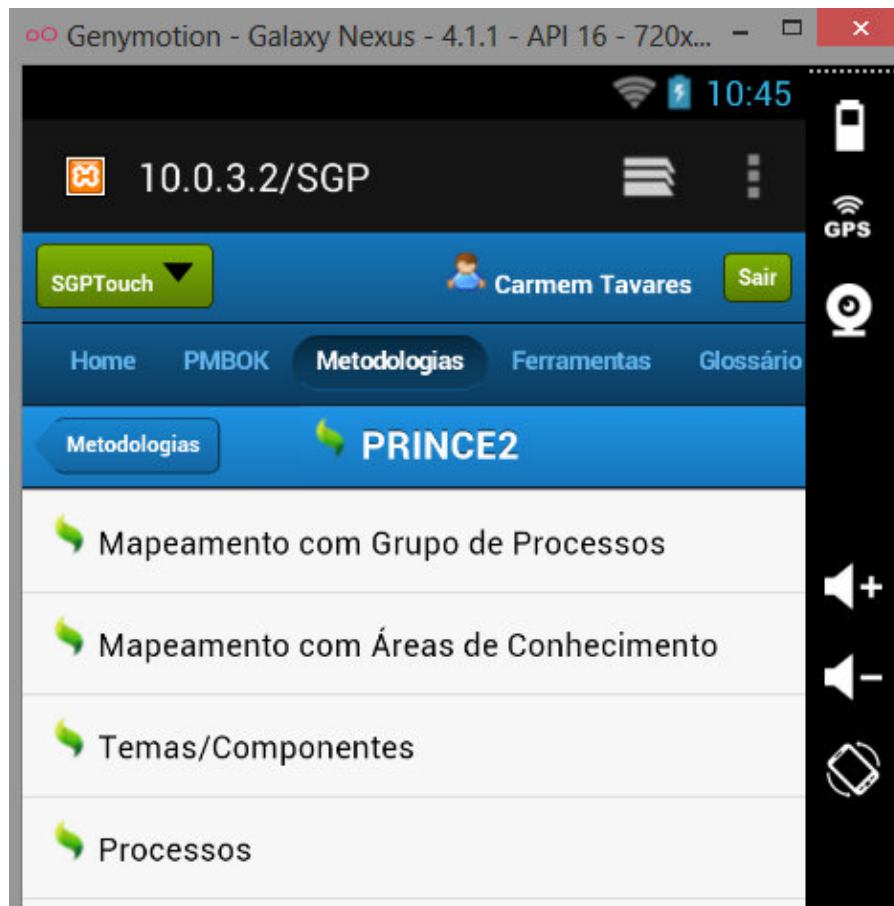


Figura 19: Menu de opções das metodologias

O utilizador ainda pode clicar no seu nome e editar os seus dados de perfil.

O Administrador tem mais duas opções (Figura 18):

- Botão “**Users**” – onde visualiza todos os utilizadores e detalhes de perfil;
- Botão “**Settings**” – onde gere toda a informação disponível.

No botão do canto superior esquerdo é possível ter acesso às seguintes opções: *About*, *Help* e *Contacto* (ver Figura 20).

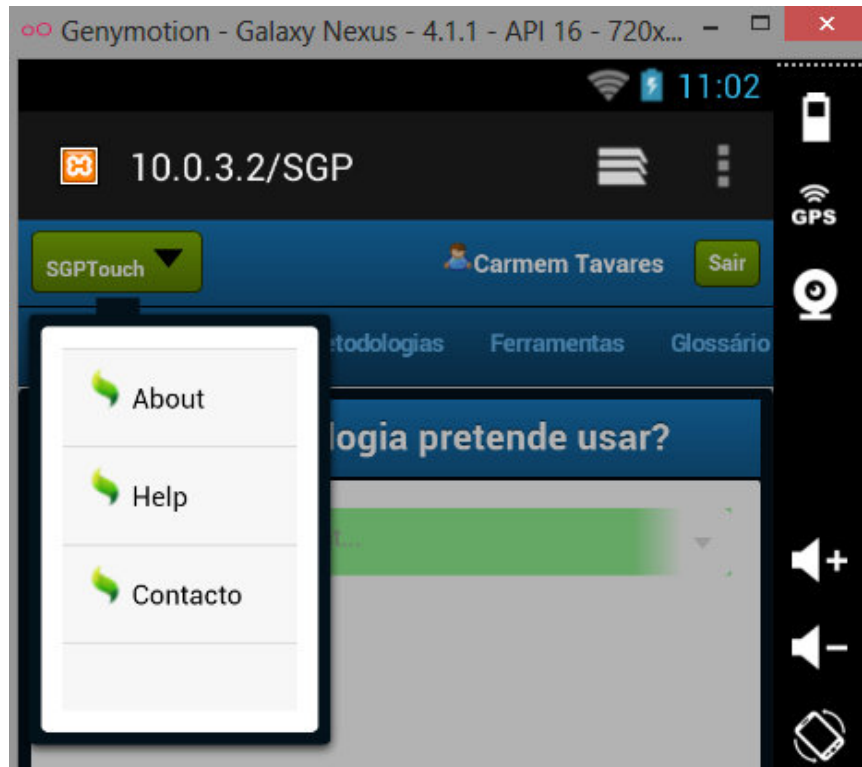


Figura 20: Outras opções (menu) da aplicação

No canto superior direito está o botão sair que termina a sessão do utilizador e redireciona a aplicação para a página de autenticação.

A Figura 21 mostra a escolha da metodologia na página inicial da aplicação num desktop. Como se pode verificar, a forma da apresentação da lista de metodologias no desktop é diferente da forma como se apresenta num dispositivo móvel (ver Figura 17).

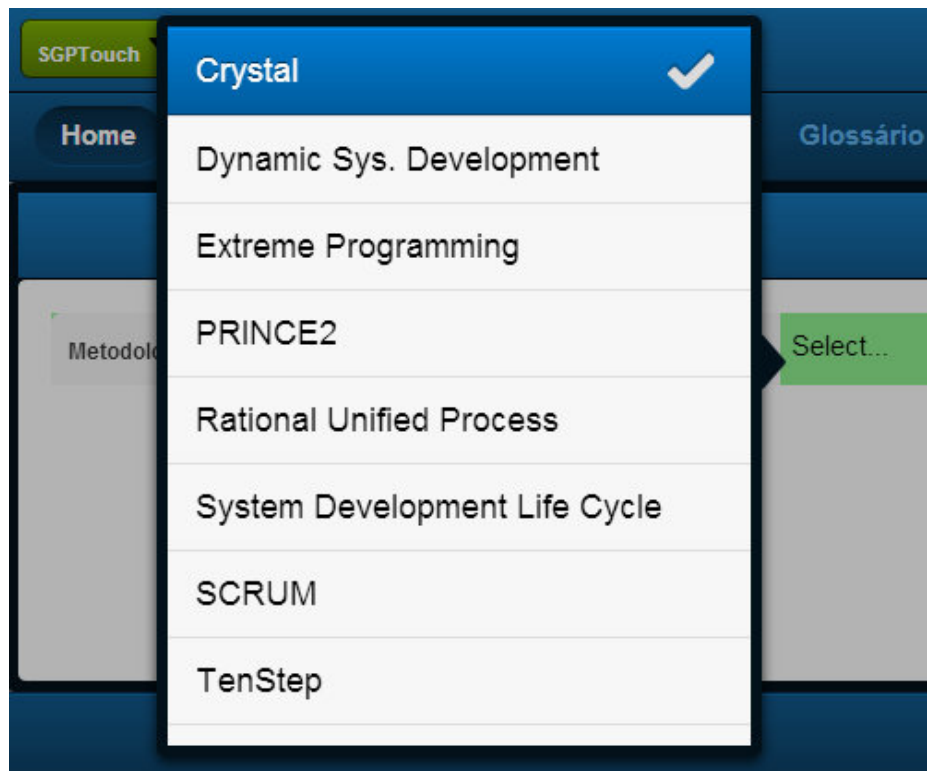


Figura 21: Visualização das metodologias na página inicial no desktop

De seguida apresenta-se um pequeno código que mostra a estrutura genérica em MVC do funcionamento da *framework* Sencha Touch. Neste caso em concreto é mostrado como funciona o processo de registo na aplicação.

Model - app/model/Registar.js

```
Ext.define('SGP.model.Registar', {
    extend: 'Ext.data.Model',
    config: {
        idProperty: 'id',
        fields: [
            {name: 'nome', type: 'string'},
            {name: 'sobrenome', type: 'string'},
            {name: 'email', type: 'string'},
            {name: 'username', type: 'string'},
            {name: 'password', type: 'string'},
            {name: 'password2', type: 'string'},
        ],
        validations: [
            { type: 'presence', field: 'email' },
            ...
            { type: 'length', field: 'password', min: 6, max: 12 },
            { type: 'email', field: 'email' },
        ]
    }
});
```

View - app/view/Registar.js

```
Ext.define('SGP.view.Registar', {
    extend: 'Ext.form.Panel',
```

```

    requires: ['Ext.form.Panel', 'Ext.field.Password', ...],
    config: {
        ...
        items:[
            { xtype: 'fieldset',
              items: [
                { xtype: 'textfield', name: 'username', label:
'Username', },
                { xtype: 'passwordfield', name: 'password',
label: 'Password'},
                ...
              ]
            },
            { xtype: 'button', text: 'Enviar', ui: 'confirm', id:
'btnRegistrar'}]
    },
});

```

Controller - app/controller/Registrar.js

```

Ext.define('SGP.controller.Registar', {
    extend: 'Ext.app.Controller',
    config: {
        models: ['Registrar'],
        refs: {
            registrar: '#btnRegistrar',
        },
        control: {
            registrar: {
                tap: 'onCreateUser'
            },
        }
    },
    onCreateUser: function() {
        ...
        if( errors.getCount()==0){
            Ext.Ajax.request({
                url: 'data/registar.php',
                params: values,
                success: function(response){
                    var text = response.responseText;
                    Ext.Msg.alert(text,'Pode autenticar.');
```

submitting the form');
 }
 });
 }
 ...
 },
});

App.js

```

Ext.Loader.setPath({
    'Ext': 'touch/src',
    'SGP': 'app'
});
Ext.application({
    name: 'SGP',
    requires: [
        'Ext.MessageBox'
    ],
    models: ['Registrar'],
    views: ['Registrar'],

```

```
    controllers: ['Registrar'],
    ...
    launch: function() {
        Ext.fly('appLoadingIndicator').destroy();
        Ext.Viewport.add(Ext.create('SGP.view.Registar'));
    },
    ...
});
```

O ficheiro **app.js** é o ficheiro principal da aplicação e contém a referência para os restantes ficheiros que compõem a aplicação. Neste ficheiro está a função **launch()** que é chamada automaticamente quando a aplicação é “lançada” e neste caso mostra o formulário de registo que se encontra no *View*. Após carregar no botão de submissão dos dados (**btnRegistrar**), o *Controller* verifica que evento deve ocorrer e neste caso é chamado a função **onCreateUser()**, que verifica o modelo de dados e faz a validação dos mesmos (*Model*), e se não houver erros comunica com o servidor enviando os dados de registo. O ficheiro **data/registar.php** trata de fazer o registo na BD e envia uma resposta que pode ser *success* em caso de sucesso ou *failure* em caso de ocorrer falha no registo.

5 Validação da Aplicação

A validação da aplicação é feita através dos diferentes casos de uso relacionando os mesmos com os requisitos. Uma validação mais extensiva e completa da aplicação seria possível através do envolvimento de profissionais de gestão de projetos. Apesar dos mesmos estarem planeados, não foi possível executá-los por restrições temporais.

5.1 Atores e Casos de Uso

A figura seguinte permite identificar os principais atores e os respetivos casos de uso (Figura 22). Existem dois atores principais, o “Administrador” e o “Utilizador”. O Administrador é quem é responsável pela gestão da aplicação e o Utilizador representa qualquer utilizador registado na mesma.

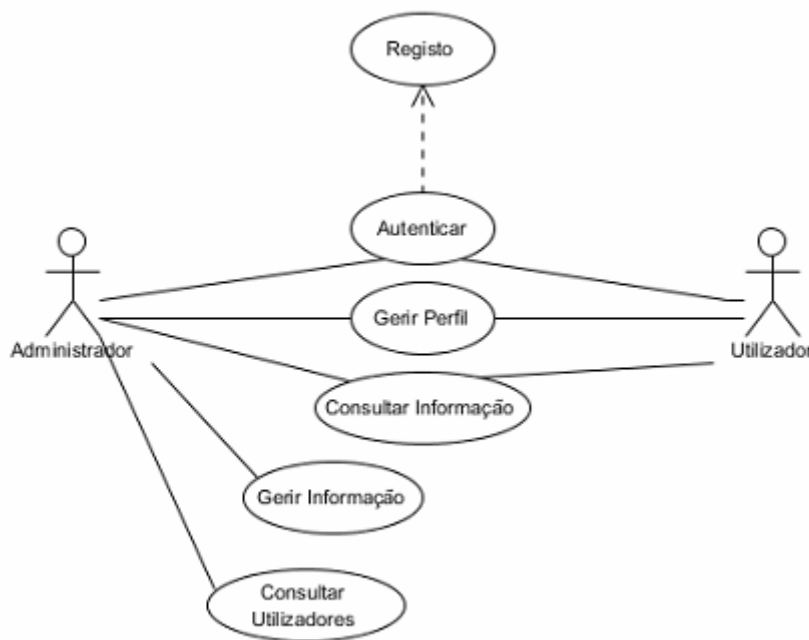


Figura 22: Atores e Casos de Uso

Os casos de uso representam as funcionalidades da aplicação na ótica dos utilizadores. A seguir são descritas os casos de uso da aplicação conforme figura em cima:

- **Autenticar** – consiste na identificação do utilizador perante o sistema.

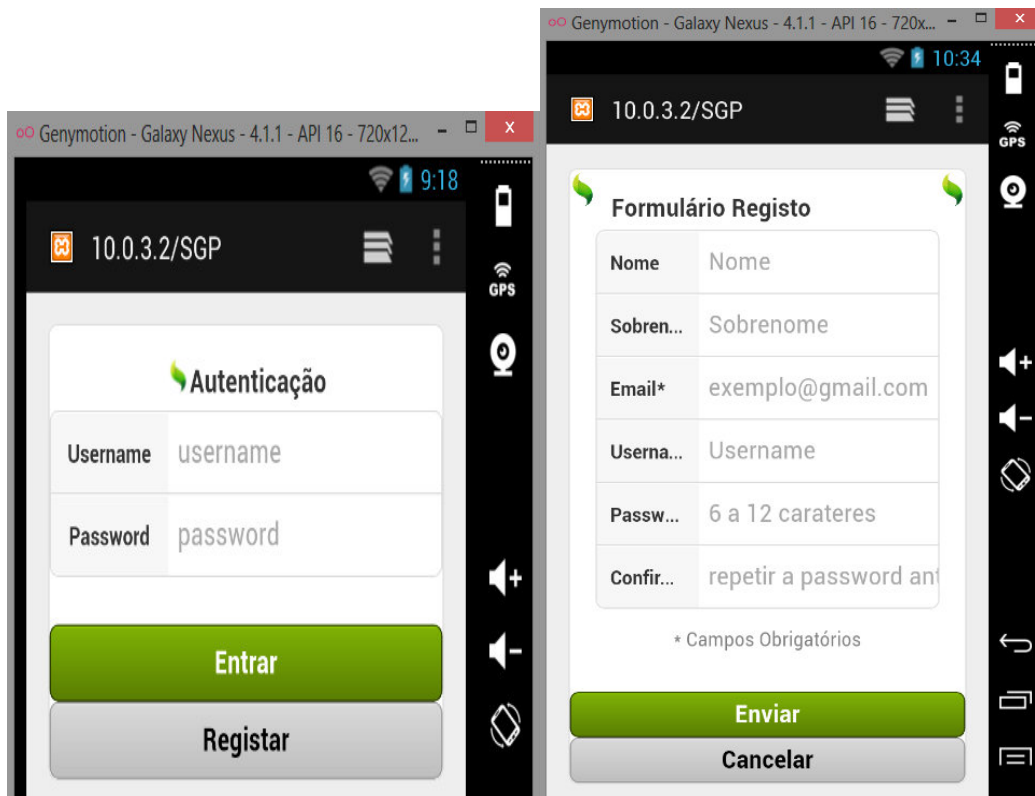


Figura 23: Ecrã de registo e autenticação

A figura anterior (Figura 23) mostra o formulário de registo (lado direito) e a janela de autenticação (lado esquerdo) da aplicação.

- **Gerir Perfil** – consiste na visualização e alteração dos dados do utilizador.

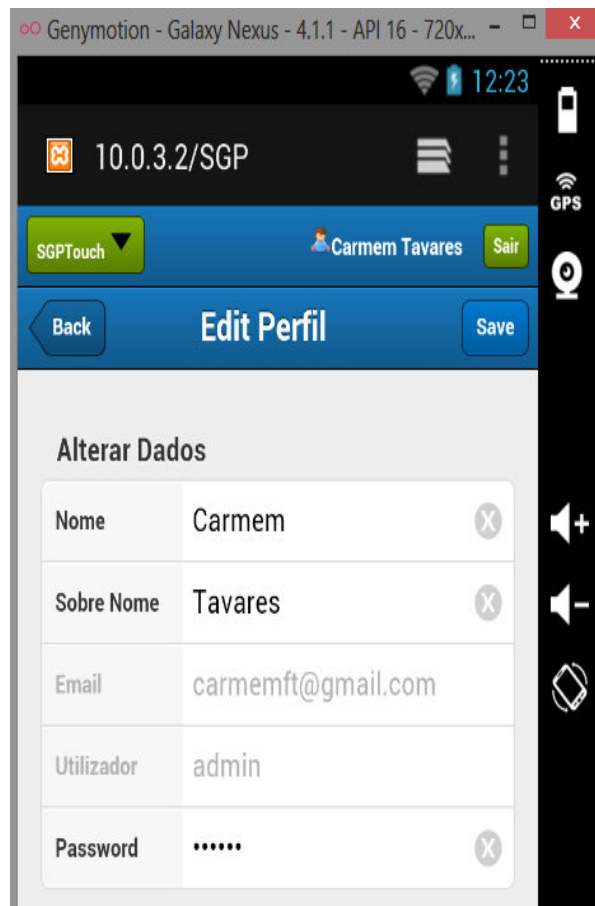


Figura 24: Visualização do perfil de utilizador

A figura anterior (Figura 24) mostra os dados do utilizador após clicar no nome do mesmo. Para gravar alterações é clicar no botão **Save** e o botão **Back** redireciona a aplicação para a página anterior. Os campos Email e Utilizador não podem ser alterados.

- **Consultar Informação** – consiste na consulta de toda a informação disponível na aplicação;

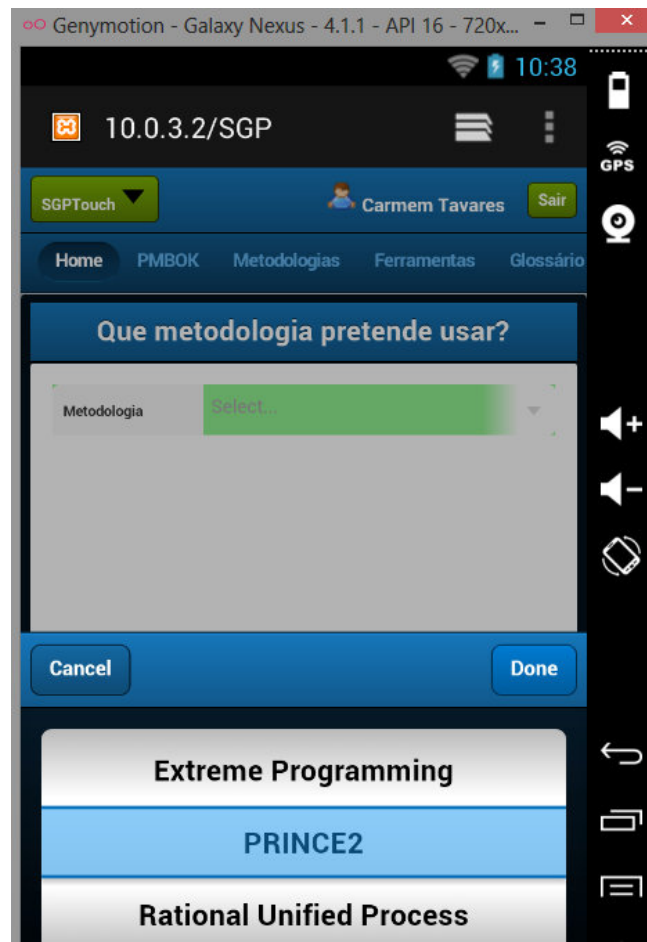


Figura 25: Ecrã inicial da aplicação

O utilizador depois de autenticar-se na aplicação é-lhe solicitado a escolha da metodologia de GP que pretende usar, onde lhe é apresentado o respetivo mapeamento com o PMBOK.

Para além disso, o utilizador tem à disposição todo um conjunto de informação relacionada com as metodologias de GP e o PMBOK.

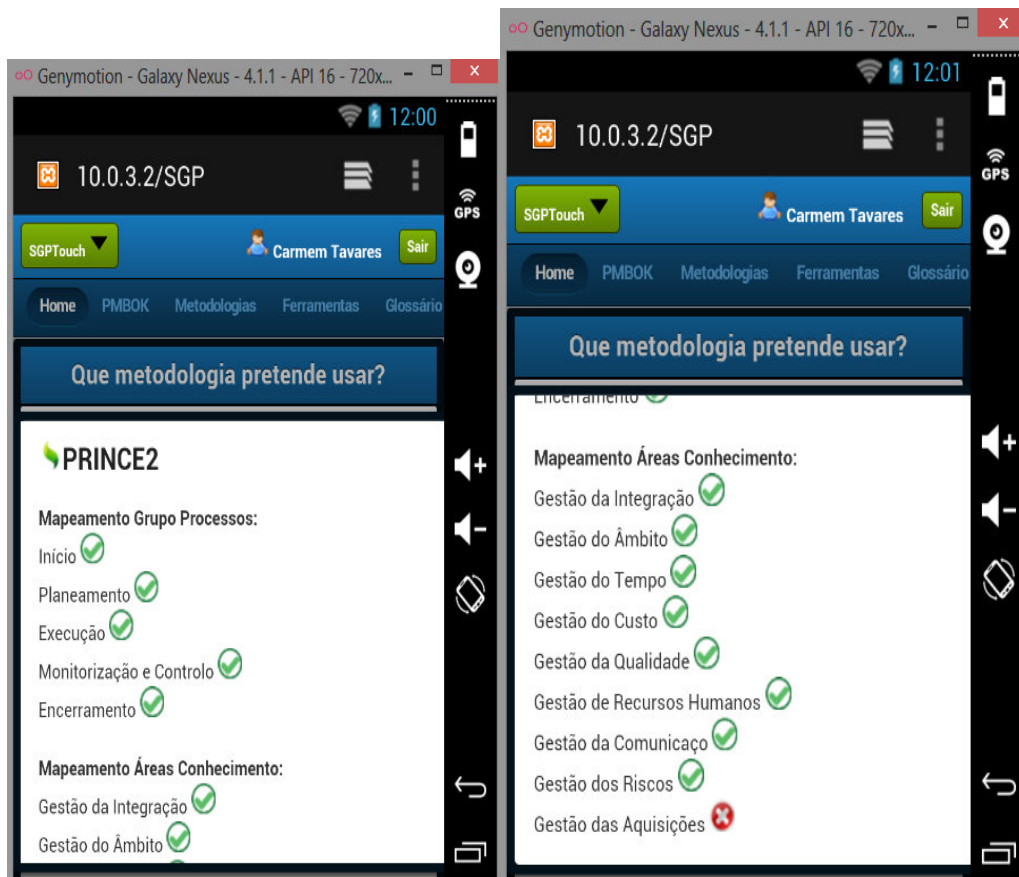


Figura 26: Visualização do mapeamento do PPRINCE2 com grupo de processos e áreas de conhecimento

Após seleccionar a metodologia pretendida é visualizado o mapeamento da mesma com os grupos de processos (lado esquerdo) e áreas de conhecimento (lado direito) conforme a figura em cima. Os itens verdes dizem que existe mapeamento e o vermelho diz o contrário.

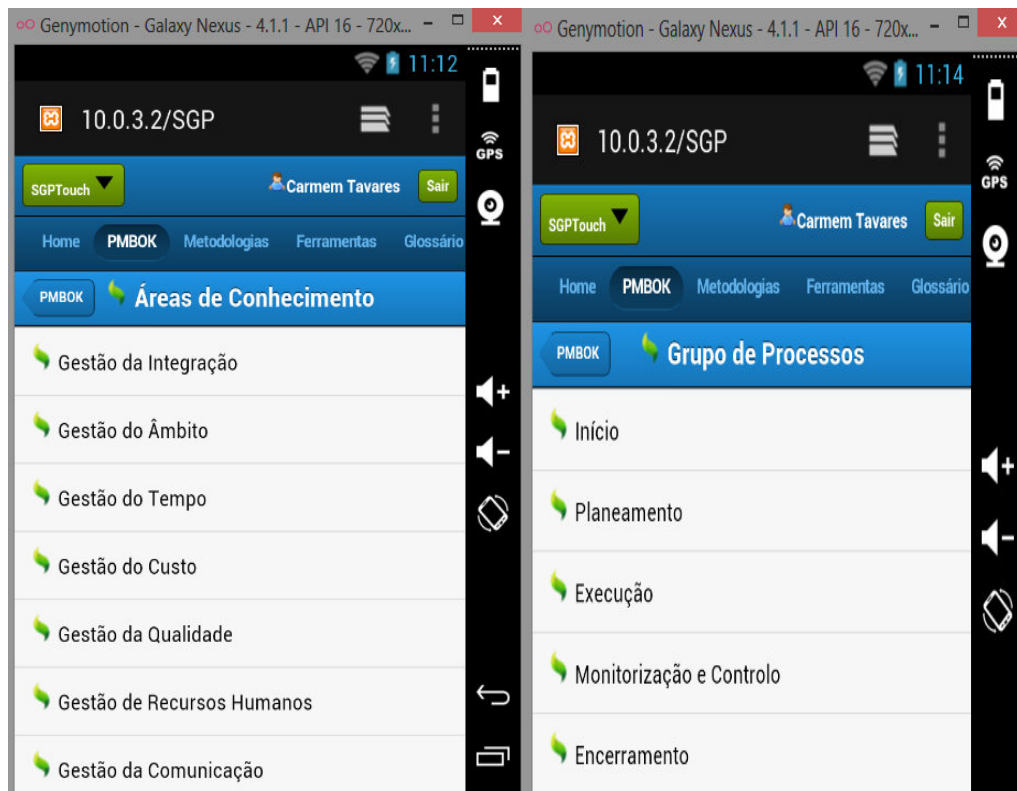


Figura 27: Visualização das áreas de conhecimento e grupo de processos

Selecionando a opção PMBOK o utilizador pode visualizar todas as áreas de conhecimento (lado esquerdo) e todos os grupos de processos (lado direito) conforme a figura em cima.

- **Gerir Informação** – consiste na disponibilização e atualização da informação no sistema;



Figura 28: Atualização dos dados no sistema

O administrador da aplicação gere toda a informação disponibilizada na aplicação. Conforme a figura anterior (Figura 28), para alterar a informação é clicar no botão **Save** (canto superior direito) e para eliminar a informação é clicar no botão **Trash** (canto inferior esquerdo). Para chegar a esta janela o administrador tem a opção **Settings**, conforme mostra a figura seguinte (Figura 29).

- **Consultar Utilizadores** – consiste na visualização de todos os utilizadores registados no sistema e os detalhes dos mesmos.

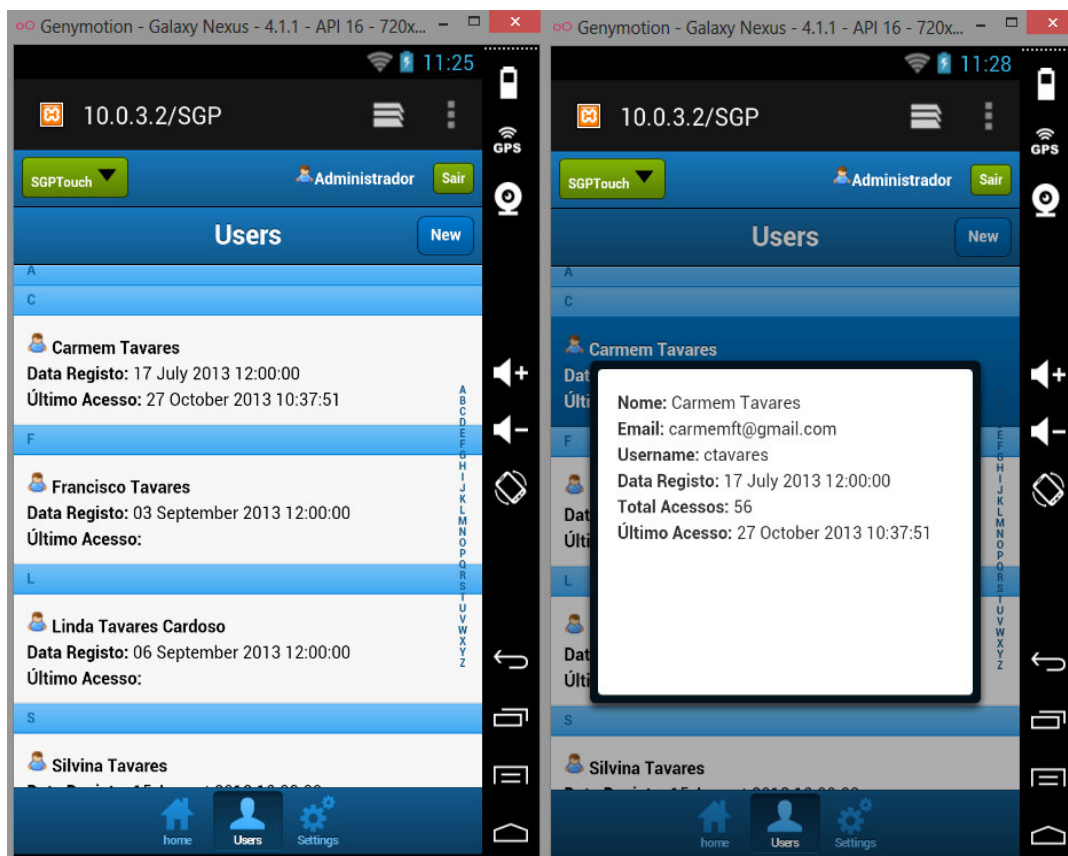


Figura 29: Visualização da lista de utilizadores e os seus dados

Ao carregar no item *Users* (menu no rodapé), é mostrada uma lista com os utilizadores que estão registados na aplicação. Esta lista está organizada de forma alfabética, onde do lado direito estão as letras organizados verticalmente e ao clicar é mostrado os utilizadores correspondentes. Ao clicar em cima do nome é mostrado um *popup* com os detalhes desse utilizador.

5.2 Validação dos Requisitos

A tabela seguinte (Tabela 15) mostra para cada caso de uso quais os requisitos é que validam.

Caso de Uso	Requisitos (ID)
Autenticar	#R02 e #R03
Gerir Perfil	#R04
Consultar Informação	De #R05 a #R12
Gerir Informação	#R15
Consultar Utilizadores	#R13 e #R14

Tabela 15: Validação dos requisitos

Cada caso de uso serviu para validar alguns dos requisitos definidos no capítulo anterior (Tabela 14).

Todos os dispositivos móveis são alvos desta aplicação e também qualquer computador *desktop* desde que utilize o browser Google Chrome ou Safari. Como foi visto no protótipo, trata-se de uma aplicação muito prática e simples de usar sem qualquer tipo de treino.

6 Conclusão e Trabalho Futuro

O presente trabalho teve como principal objetivo alinhar algumas das principais metodologias de gestão e de desenvolvimento de projetos de sistemas de informação (inclusive algumas ágeis) com as boas práticas recomendadas pelo PMBOK. Por outro lado, um outro dos objetivos do trabalho consistiu no desenvolvimento de uma aplicação web/mobile que permitisse que os gestores de projeto pudessem ter um apoio significativo no alinhamento dos seus projetos com os processos e áreas de conhecimento do PMBOK. Esta ferramenta servia como uma ferramenta de apoio para o gestor de projeto perceber como poderia alinhar as suas metodologias de desenvolvimento e gestão de projetos de sistemas de informação com as práticas normalizadas do PMBOK.

Assim, este trabalho demonstrou que as práticas do PMBOK são perfeitamente compatíveis com as disciplinas e fases do Rational Unified Process (RUP), com os processos e temas/componentes do PRINCE2 assim como também com as práticas/valores das metodologias ágeis de desenvolvimento de software - neste caso Extreme Programming (XP) e SCRUM. A arte da sua integração na gestão de projetos, está em saber adaptar o processo para o projeto em questão através do conhecimento sobre as metodologias e de como utilizá-las na sua implementação. A ferramenta desenvolvida ajudará os gestores de projetos neste mesmo processo.

Cada situação requer um ponto de equilíbrio que deve ser usada de forma apropriada no processo de integração das práticas das metodologias com as práticas do PMBOK. O PMBOK não proíbe o uso das suas práticas com outras metodologias. Porém, requer que o gestor do projeto tenha conhecimento de ambas as práticas. Ambos abordam práticas que caminham para o mesmo objetivo, ou seja, entregar o produto dentro do prazo e custo estabelecido e com qualidade, cumprindo as expectativas do cliente. Diferem nas práticas e na forma como as empregam, por isso, nada os impedem de ajustar-se de modo a melhorar a eficácia no alcance do objetivo.

De referir que no seio da pesquisa para a elaboração deste trabalho não foi encontrada nenhuma publicação que fizesse simultaneamente o mapeamento do PMBOK com as quatro metodologias: RUP, PRINCE2, XP e SCRUM. Por outro lado, não existia

igualmente nenhuma ferramenta que oferecesse suporte para os gestores de projetos poderem efetuar esse mesmo mapeamento, assim como informação adicional sobre as tarefas a executar em cada uma das fases e processos do mapeamento.

A aplicação web/mobile desenvolvida, permite ter acesso a toda a informação em relação ao PMBOK e às metodologias aqui referidas, de uma forma integrada. A informação existente é muito desagregada e não permite um acesso rápido, interativo e facilitado à mesma, pelo que a aplicação desenvolvida serve de apoio à decisão do gestor de projeto – em termos de informação da metodologia usada e do correspondente mapeamento no PMBOK.

Com estas informações concentradas num único ponto e com uma forma cómoda, prática e interativa de acesso aos mesmos, por parte de qualquer elemento da equipa do projeto, espera-se que esta aplicação possa constituir uma mais-valia e uma forma adicional de combater certas lacunas que contribuem para as falhas que ocorrem no decorrer da execução dos projetos de desenvolvimento de SI.

Infelizmente, não foi possível, por manifesta falta de tempo, efetuar uma validação mais completa recorrendo a um conjunto de gestores de diversos tipos de projetos de desenvolvimento de sistemas de informação que recorressem a múltiplas metodologias. Esta validação teria sido importante pois iria permitir recolher informação valiosa que poderia ser utilizada na melhoria da própria ferramenta.

No decurso deste trabalho, e com o aproximar da sua conclusão foi possível identificar um conjunto de aspetos que poderiam contribuir para a melhoria e engrandecimento da solução concebida, e que podem facilmente ser apontados como trabalho futuro.

Um desses trabalhos futuros seria a análise/mapeamento das metodologias em relação às entradas, ferramentas e técnicas, e saídas dos processos do PMBOK – para cada processo averiguar quais são as entradas, ferramentas e técnicas que devem ser aplicadas e quais são as saídas resultantes, conforme a Figura 10, que se apresenta no Capítulo 3 deste trabalho.

Por outro lado seria igualmente interessante, a evolução da aplicação desenvolvida para uma ferramenta de gestão de projetos. Ou seja, em vez de ser uma aplicação apenas de consulta de informação, seria acrescentada a componente específica de gestão de projetos de SI, à semelhança das ferramentas descritas no Capítulo 2.

Finalmente, uma outra sugestão consistiria na evolução a aplicação para ajudar os gestores de projeto na tomada de decisão – uma vez evoluída a aplicação com a componente de gestão de projetos, fazer com que o gestor de projeto seja alertado nas decisões a tomar de acordo com o andamento do projeto, metodologia adotada e práticas/princípios dessa metodologia.

Em jeito de conclusão final, importa frisar que os resultados deste meu trabalho serão igualmente alvo de uma publicação científica que está em preparação (no momento em que este texto foi escrito).

Referências

- Accenture, “The Accenture CIO Mobility Survey 2012”,
<http://www.accenture.com/SiteCollectionDocuments/PDF/Accenture-CIO-Mobility-Survey-2012-v2.pdf> (consultado em 27 Dez 2012), Dez 2012.
- AceProject, “Feature-rich ... without the Complexity”, <http://www.aceproject.com/>
(consultado em 28 Jan 2013), Jan 2013.
- Basecamp, <https://basecamp.com/> (consultado em 27 Jan 2013), Jan 2013
- Beck, Kent, “Extreme Programming Explained: Embrace Change”, Addison – Wesley, New York, NY, 1999.
- Beck, Kent, and Fowler, Martin, “Extreme Programming Explained”. Addison Wesley: ISBN 0-201-71091-9, 2000.
- Charbonneau, Serge, “Software Project Management - A Mapping between RUP and the PMBOK”. Rational Edge, [s.l.], 15 May 2004.
http://download.boulder.ibm.com/ibmdl/pub/software/dw/rationaledge/may04/TheRationalEdge_May2004.pdf (consultado em 20 Set 2013), Set 2013.
- Charvat, Jason, “Project Management Methodologies (A book review by R. Max Wideman)”, published by Wiley, NJ, 2003,
http://www.maxwideman.com/papers/pm-methodologies/pm_methodologies.pdf
(consultado em 26 Fev 2012), Fev 2012.
- Conlin, J. and Retik, A., “The applicability of project management software and advanced IT techniques in construction delays mitigation, International Journal of Project Management”, vol. 15, nº 2, 1997.
- Coronado, Sergio, “PMBOK (PMP) and PRINCE2 (Practitioner) Face-to-Face”, 2008,
<http://andradeivan.com/wp-content/uploads/2012/04/PMBOK-Face-to-Face-Against-PRINCE2.pdf> (consultado em 20 Set 2013), Set 2013.
- DO.Com, <https://do.com/> (consultado em 28 Jan 2013), Jan 2013.
- DotProject (2013a), “DotProject”, <http://pt.wikipedia.org/wiki/DotProject> (consultado em 27 Jan 2013), Jan 2013.
- DotProject (2013b), “welcome to dotProject.net”, <http://www.dotproject.net/>
(consultado em 27 Jan 2013), Jan 2013.
- English, Arthur, “Software Project Management – Leveraging RUP, OpenUP, and the PMBOK”,
https://www.pmiwdc.org/sites/default/files/presentations/201302/PMIW_PMTols_Presentation_LeveragingRUPOpenUPandPMBOK.pdf (consultado em 20 Set 2013), Set 2013.
- Ganttlic, “Principais Funcionalidades”, <http://www.ganttlic.com/> (consultado em 27 Jan 2013), Jan 2013.

Referências

- Goebel, Clement, “Extreme Programming Practices Used to Facilitate Effective Project Management”, Menlo Institute LCC, 2003.
- IBM (2012a), “Rational Unified Process – Best practices for software development teams”, <http://www.ibm.com/software/awdtools/rup/> (consultado em 15 Ago 2012), Ago 2012.
- IBM (2012b), ftp://public.dhe.ibm.com/software/rational/web/datasheets/RUP_DS.pdf (consultado em 15 Ago 2012), Ago 2012.
- Ionline, “Impacto da Mobilidade”, <http://www.ionline.pt/opiniao/impacto-da-mobilidade> (consultado em 27 Dez 2012), Dez 2012.
- Jalote, Pankaj, “Software Project Management in Praticce”, Addison-Wesley Professional, Reading, MA, 2002.
- Ken, Schwaber, and Jeff, Sutherland, “O Guia do Scrum”, 2011, <https://www.scrum.org/Scrum-Guides> (consultado em 21 Ago 2012), Ago 2012.
- Kerzner, Harold, “Project Management – A Systems Approach to Planning, Scheduling, and Controlling”, Ninth Edition, Jonh Wiley & Sons, Inc., New York, NY, 2005.
- Microsoft, “Novidades no Project 2013”, <http://office.microsoft.com/pt-pt/project-help/novidades-no-project-2013-HA102749523.aspx> (consultado em 27 Jan 2013), Jan 2013.
- Miguel, António, “ Gestão Moderna de Projetos Melhores Técnicas e Práticas”, 6ª Edição, FCA, 2009.
- Miguel, António, “ Gestão de Projetos de Software”, 6ª Edição Actualizada, FCA, 2010.
- Murray, Andy, “PRINCE2 in one thousand words”, 2011, http://www.best-management-practice.com/gempdf/prince2_in_one_thousand_words.pdf (consultado em 20 Set 2013), Set 2013.
- OpenWorkbench, <http://www.openWorkbench.org> (consultado em 27 Jan 2013), Jan 2013.
- Pinto, Miguel, “Gestão de Projetos com Processos Ágeis”, Instituto Superior Técnico, 2010. Dissertação de Mestrado. <https://dspace.ist.utl.pt/bitstream/2295/748756/1/> (consultado em 28 Jan 2013), Jan 2013.
- Pivotal Tracker, “Features”, <http://www.pivotaltracker.com/> (consultado em 28 Jan 2013), Jan 2013.
- Pressman, Roger.S., “Software Engineering – A Practitioner´s Approach”, McGraw-Hill, Sixth Edition, 2005.
- PRINCE2 (2012a), “What is PRINCE2?”, <http://www.prince2.com/what-is-prince2.asp> (consultado em 21 Ago 2012), Ago 2012.
- PRINCE 2 (2012b), “PRINCE2 Process Model”, <http://www.prince2training-uk.org/prince2-process-model/> (consultado em 21 Ago 2012), Ago 2012.
- Project Management Institute, “A Guide to the Project Management Body of Knowledge (PMBOK Guide)”, Fourth Edition - Ed. 2008.
- RallyDev, “What is Rally”, <http://www.rallydev.com/> (consultado em 28 Jan 2013), Jan 2013.

Referências

- Redmine, “Wiki”, <http://www.redmine.org/> (consultado em 27 Jan 2013), Jan 2013
- Roldão, Victor, “Gestão de Projectos – Uma Perspectiva Integrada”, 1ª Edição, 2000.
- Roldão, Victor, “Gestão de Projectos – Abordagem Instrumental ao Planeamento, Organização e Controlo”, 1ª Edição, 2005.
- Schwaber, Ken, and Beedle, Mike, “Agile Software Development with SCRUM”. Prentice Hall, 2002.
- Sencha Touch, <http://www.sencha.com/> (consultado em 28 Set 2013), Set 2013.
- Silva, Márcio, “Microsoft Office Project 2007, Depressa e Bem, 3ª Edição, FCA, 2007
- Silva, Alberto e Videira, Carlos, “UML, Metodologias e Ferramentas CASE”, 2ª Edição, Volume 2, Março 2008.
- Smashingapps, “10 Free Tools for Effective Project Management”, <http://www.smashingapps.com/2010/02/15/10-free-tools-for-effective-project> (consultado em 27 Jan 2013), Jan 2013.
- Sutherland, Jeff, and Ahmad, Nafis, “How a Traditional Project Manager Transforms to Scrum”, Salt Lake City, 2011, http://www.agilealliance.org/files/session_pdfs/Presentation%20-%20How%20a%20Traditional%20Project%20Manager%20Transforms%20to%20Scrum%20-%20FINAL.pdf (consultado em 20 Set 2013), Set 2013.
- TargetProcess, “Product”, <http://www.targetprocess.com/> (consultado em 28 Jan 2013), Jan 2013.
- Tek, “Taxa de penetração dos smartphones em Portugal aumenta pela sétima vez consecutiva”, http://tek.sapo.pt/tek_mobile/equipamentos/taxa_de_penetracao_dos_smartphones_em_portuga_1334895.html (consultado em 10 Set 2013), Set 2013
- TenStep (2012a), “Os princípios da metodologia Tenstep”, <http://tenstep.pt/os-principios-da-metodologia-tenstep/> (consultado em 08 Out 2012), Out 2012.
- TenStep (2012b), “Tenstep Process Model”, <http://www.tenstep.com/open/A5TSPProcessModel.html> (consultado em 08 Out 2012), Out 2012.
- VersionOne, “Product Feature”, <http://www.versionone.com//> (consultado em 28 Jan 2013), Jan 2013.
- WKP, <http://pt.wikipedia.org/wiki/Scrum> (consultado em 21 Ago 2012), Ago 2012.

Anexos

Anexo A – Detalhes dos Processos do PMBOK Guide

4 Gestão da Integração
4.1 Desenvolvimento do Project Charter – documento que autoriza formalmente o projeto ou uma fase e documenta os requisitos iniciais que satisfazem as necessidades e expectativas dos <i>stakeholders</i> ;
4.2 Desenvolvimento do Plano de Gestão de Projeto - definir e documentar as ações necessárias para definir, integrar e coordenar todos os planos subsidiários, num plano de gestão de projetos;
4.3 Direção e Gestão da Execução do Projeto – realizar o trabalho definido no plano do projeto, para satisfazer os objetivos do projeto;
4.4 Monitorização e Controlo do Trabalho do Projeto – monitorizar, rever e regular o progresso do projeto para satisfazer os objetivos de desempenho definidos no plano do projeto;
4.5 Controlo Integrado da Mudança – rever todos os pedidos de alteração, aprovar e gerir as alterações às entregas, ativos organizacionais, documentos do projeto e plano do projeto;
4.6 – Encerrar o Projeto ou fase – concluir todas as atividades dos grupos de processos da gestão do projeto para concluir formalmente o projeto ou uma fase.

5 Gestão do Âmbito
5.1 Recolher os Requisitos – definir e documentar as necessidades dos <i>stakeholders</i> de modo a satisfazer os objetivos do projeto;
5.2 Definir o Âmbito – desenvolver uma descrição detalhada do produto e do projeto;
5.3 Criar o WBS – subdividir as entregas e o trabalho do projeto em componentes menores e melhor geríveis;
5.4 Verificar o Âmbito – formalizar a aceitação das entregas concluídas do projeto;
5.5 Controlar o Âmbito – monitorizar a situação do âmbito do projeto e controlar as alterações à <i>baseline</i> do âmbito.

6 Gestão do Tempo
6.1 Definir Atividades – identificar as atividades a serem realizadas para produzir as diversas entregas do projeto;
6.2 Sequenciar Atividades – identificar e documentar as dependências entre as várias atividades;
6.3 Estimar Recursos das Atividades – estimar o tipo e quantidade de recursos necessários para realizar cada atividade;
6.4 Estimar Duração das Atividades – estimar o número de períodos de trabalho necessários para concluir as atividades individuais, com recursos estimados;
6.5 Criar Calendarização – analisar sequências duração, necessidades de recursos e restrições temporais das atividades para criar o cronograma do projeto;
6.6 Controlar a Calendarização – monitorizar a situação do projeto para atualizar o progresso e controlar as alterações ao cronograma.

7 Gestão do Custo
7.1 Estimar Custos - desenvolver uma aproximação dos custos dos recursos necessários para concluir as atividades do projeto;
7.2 Determinar o Orçamento - agregar os custos estimados para as atividades ou pacotes de trabalho individuais para estabelecer a <i>baseline</i> do custo;
7.3 Controlar Custos – monitorizar a situação do projeto para atualizar o orçamento e gerir as alterações à <i>baseline</i> do custo

8 Gestão da Qualidade
8.1 Planear a Qualidade – identificar os requisitos e/ou <i>standards</i> da qualidade relevantes para o produto e o projeto e documentar o modo como o projeto irá satisfazer;
8.2 Realizar a Verificação da Qualidade – auditar os requisitos da qualidade e os resultados das medições do controlo da qualidade para assegurar que são usados os padrões de qualidade e as definições operacionais adequados;
8.3 Controlar a Qualidade – monitorar resultados específicos do projeto para determinar se satisfazem os padrões relevantes da qualidade e identificar formas e eliminar causas de desempenho insatisfatório.

9 Gestão de Recursos Humanos
9.1 Desenvolver o Plano de Recursos Humanos – identificar e documentar os papéis, responsabilidades e relações hierárquicas dos recursos humanos do projeto;
9.2 Constituir a Equipa de Projetos - obter os recursos humanos necessários;
9.3 Desenvolver a Equipa de Projeto – melhorar as competências e a interação dos membros da equipa para melhorar o desempenho do projeto;
9.4 Gerir a Equipa de Projeto – monitorar o desempenho dos membros da equipa fornecer informação, resolver problemas e coordenar alterações para melhorar o desempenho do projeto.

10 Gestão da Comunicação
10.1 Identificar os Stakeholders – identificar todas as pessoas e organizações impactadas pelo projeto e documentar informação relevante acerca dos seus interesses, envolvimento e impacto no sucesso do projeto;
10.2 Planear as Comunicações – determinar as necessidades de informação e comunicação dos <i>stakeholders</i> do projeto;
10.3 Distribuir a Informação – disponibilizar a informação necessária de um modo oportuno;
10.4 Gerir as Expectativas dos Stakeholders – comunicar e trabalhar com os <i>stakeholders</i> para satisfazer as suas necessidades e resolver eventuais problemas;
10.5 Relatar o Desempenho – recolher e distribuir a informação sobre o desempenho do projeto, incluindo relatórios de situação medidas do progresso e previsões.

11 Gestão dos Riscos
11.1 Planear a Gestão do Risco – decidir o modo de abordar, planear e executar as atividades de gestão do risco para o projeto;
11.2 Identificar Riscos – determinar os riscos do projeto e documentar as suas características;
11.3 Análise Qualitativa dos Riscos – priorizar os riscos para análise ou ação

subsequente, através da avaliação e combinação das respetivas probabilidades de ocorrência e impacto;
11.4 Análise Quantitativa dos Riscos – analisar numericamente o efeito dos riscos identificados sobre os objetivos do projeto;
11.5 Planear Respostas a Riscos – desenvolver ações para melhorar as oportunidades e reduzir as ameaças aos objetivos do projeto;
11.6 Monitorar os Riscos – monitorizar os riscos identificados, identificar novos riscos, executar os planos de respostas e avaliar a sua eficácia.

12 Gestão da Aquisições
12.1 Planear Aquisições – documentar as decisões de aquisições para o projeto, especificar a estratégia e abordagem e identificar potenciais fornecedores;
12.2 Conduzir Aquisições – obter respostas dos fornecedores, selecionar o (s) fornecedor (es) e assinar o (s) contrato (s);
12.3 Gerir Aquisições – gerir as relações contratuais das diversas aquisições, monitorar o desempenho do (s) contrato (s) e efetuar as necessárias correções e alterações;
12.4 Fechar Aquisições – concluir e encerrar cada contrato, a resolução de questões em aberto.

Anexo B – Mapeamento detalhado entre o RUP e o PMBOK

4 Gestão da Integração	RUP
4.1 Desenvolvimento do Project Charter	Desenvolver o modelo de negócio, aprovar a revisão do projeto e iniciar o projeto
4.2 Desenvolvimento do Plano de Gestão de Projeto	Planear fases e iterações, desenvolver plano de métricas, desenvolver plano de resolução de problemas, desenvolver plano de iteração, e plano de aceitação do produto e compilar plano de desenvolvimento de software
4.3 Direção e Gestão da Execução do Projeto	Programar e atribuir trabalho, controlar exceções e problemas, definir processos de controlo e monitoramento, revisão da aceitação da iteração e do plano de iteração, iniciar iteração
4.4 Monitorização e Controlo do Trabalho do Projeto	Monitorização do estado do projeto, relatório de estado, <i>Project Review Authority</i> (PRA) e revisão do projeto
4.5 Controlo Integrado da Mudança	Controlado pelas disciplinas de configuração e gestão de mudança do RUP
4.6 Encerramento do Projeto ou Fase	Preparar o fecho da fase, revisão do ciclo de vida e <i>milestone</i> , preparar o fecho do projeto e revisão da aceitação do projeto

5 Gestão do Âmbito	RUP
5.1 Recolher os Requisitos	Controlado pela disciplina de requisitos
5.2 Definir o Âmbito	Controlado pela disciplina de requisitos
5.3 Criar o WBS	Sem mapeamento direto
5.4 Verificar o Âmbito	Revisão do ciclo de vida e <i>milestone</i> do

Anexos

	projeto
5.5 Controlar o Âmbito	Avaliar iteração e planejar fases e iterações

6 Gestão do Tempo	RUP - Tarefas
6.1 Definir Atividades	Planejar fases e iterações e desenvolver plano de iteração
6.2 Sequenciar Atividades	Planejar fases e iterações e desenvolver plano de iteração
6.3 Estimar Recursos das Atividades	Planejar fases e iterações e desenvolver plano de iteração
6.4 Estimar Duração das Atividades	Planejar fases e iterações e desenvolver plano de iteração
6.5 Criar Calendarização	Planejar fases e iterações e desenvolver plano de iteração
6.6 Controlar a Calendarização	Avaliar iteração, controlar exceções e problemas, revisão do planeamento do projeto

7 Gestão do Custo	RUP
7.1 Estimar Custos	Planejar fases e iterações
7.2 Determinar o Orçamento	Sem mapeamento
7.3 Controlar Custos	Sem mapeamento

8 Gestão da Qualidade	RUP
8.1 Planejar a Qualidade	Desenvolver plano de garantia de qualidade
8.2 Realizar a Verificação da Qualidade	Controlado pelas disciplinas de configuração e gestão de mudança do RUP
8.3 Controlar a Qualidade	Controlado pelas disciplinas de configuração e gestão de mudança do RUP

9 Gestão de Recursos Humanos	RUP
9.1 Desenvolver o Plano de Recursos Humanos	Definir a organização do projeto e as pessoas
9.2 Constituir a Equipa de Projetos	Adquirir pessoas
9.3 Desenvolver a Equipa de Projeto	Sem mapeamento
9.4 Gerir a Equipa de Projeto	Programar e atribuir tarefas, controlo de exceções e problemas

10 Gestão da Comunicação	RUP
10.1 Identificar os <i>Stakeholders</i>	Controlado pelas disciplinas de modelo de negócios e requisitos
10.2 Planejar a Comunicação	Compilar plano de desenvolvimento de software
10.3 Distribuir Informação	Relatório de estado
10.4 Gerir as Expectativas dos <i>Stackholders</i>	Planejar fases e iterações
10.5 Relatar o Desempenho	Relatório de estado

11 Gestão dos Riscos	RUP
11.1 Planear a Gestão de Risco	Desenvolver plano de gestão de riscos
11.2 Identificar Riscos	Identificar e avaliar riscos
11.3 Análise Qualitativa de Risco	Identificar e avaliar riscos
11.4 Análise Quantitativa de Risco	Identificar e avaliar riscos
11.5 Planear Respostas a Risco	Identificar e avaliar riscos
11.6 Monitorar os Riscos	Identificar e avaliar riscos

12 Gestão da Aquisições	RUP
12.1 Planear Aquisições	Sem mapeamento
12.2 Conduzir Aquisições	Sem mapeamento
12.3 Gerir Aquisições	Sem mapeamento
12.4 Fechar Aquisições	Sem mapeamento

Anexo C – Mapeamento detalhado entre o SCRUM e o PMBOK

4 Gestão da Integração	SCRUM - Processos/prática
4.1 Desenvolvimento do Project Charter	<i>Product Owner</i> e a equipa Scrum desenvolvem <i>Product Roadmap, Vision e Backlog</i>
4.2 Desenvolvimento do Plano de Gestão de Projeto	Equipa Scrum desenvolve um plano de entrega (<i>release</i>) de alto nível e um plano mais detalhado para o próximo <i>Sprint</i> (<i>Rolling wave Planning</i>)
4.3 Direção e Gestão da Execução do Projeto 4.4 Monitorização e Controlo do Trabalho do Projeto	Equipa Scrum executa e entrega; Scrum Master gere os princípios do Scrum, que por sua vez gerem as equipas. Equipa Scrum se auto-gere usando revisões do <i>Sprint</i> e faz ajustes de modo a obter melhoria contínua
4.5 Controlo Integrado da Mudança	<i>Product Owner</i> e a equipe Scrum controlam a mudança através do <i>Product Backlog</i> , <i>feedback</i> constante durante a iteração e revisão
4.6 Encerramento do Projeto ou Fase	Comentário do <i>Sprints Retrospective</i> do projeto; <i>Sprint</i> N+1 para o encerramento administrativo ou auditorias, se necessário

5 Gestão do Âmbito	SCRUM - Processos/prática
5.1 Recolher os Requisitos	Desenvolver e priorizar os itens do <i>Product Backlog</i>
5.2 Definir o Âmbito	Selecionar os itens do <i>Product Backlog</i> para entrega ou <i>Sprints</i>
5.3 Criar o WBS	Criar a <i>Feature Breadown Struture</i> (FBS) para entregas, mostrando as características de cada entrega. Além disso decompô-lo em características individuais (cenários) por

Anexos

	<i>Sprint</i>
5.4 Verificar o Âmbito	Aceitação do FBS por <i>Product Owner</i>
5.5 Controlar o Âmbito	Gestão por <i>Product Backlog</i> e <i>Product Owner</i> ; proteção da iteração

6 Gestão do Tempo	SCRUM - Processos/prática
6.1 Definir Atividades	Recursos são selecionados para um <i>Sprint</i> por equipa; tarefas são identificadas para cumprir os requisitos
6.2 Sequenciar Atividades 6.3 Estimar Recursos das Atividades 6.4 Estimar Duração das Atividades	Conduzido pela equipa durante as reuniões de planeamento de <i>Sprint</i> ; estimativa de tarefas para completar uma história
6.5 Criar Calendarização	Uma agenda de divulgação é desenvolvida; apenas os recursos direcionados para os <i>Sprints</i> são elaborados e estimado (<i>Justi-In-Time Planning</i>)
6.6 Controlar a Calendarização	Equipa gere as tarefas a serem realizadas no <i>Sprint</i>

7 Gestão do Custo	SCRUM - Processos/prática
7.1 Estimar Custos	Realizar estimativa top-down das entregas e <i>Sprints</i> , usando <i>Velocity Project</i> , dias ideais, analogia, opinião do especialista ou desagregação. Realizar uma estimativa bottom-up do <i>Sprint</i> em questão para validar ou afinar as estimativas <i>top-down</i> .
7.2 Determinar o Orçamento	Criar uma linha de custos base; rever a linha de custo base depois de alguns <i>Sprints</i> (quando a velocidade da equipe atual é conhecida)
7.3 Controlar Custos	Uso do <i>Product Burndown Chart</i> como um assessor de controlo de custos; usar <i>Agile EVM</i> em ambientes mais formais

8 Gestão da Qualidade	SCRUM - Processos/prática
8.1 Planear a Qualidade	Qualidade é implícita por meio de práticas Scrum (definição de metas, testes e frequente, software de trabalho, remoção de impedimento, codificação / testes padrões, métricas). Qualidade é responsabilidade de toda a equipa multifuncional do Scrum
8.2 Realizar a Verificação da Qualidade	Geralmente, realizada pela equipa. Em ambientes formais, uma terceira parte pode ser contratado para garantir a qualidade como parte de <i>Sprint</i> adicional (<i>Sprint</i> N +1) para cumprir os requisitos regulamentares e de conformidade

8.3 Controlar a Qualidade	Realizado pela própria equipa usando testes de unidade, integração e testes de funcionalidade pela equipa de teste e testes de aceitação do utilizador (proprietário do produto) e testes automatizados; Usar gráficos burndown para monitorar as tendências de desenvolvimento.
---------------------------	--

9 Gestão de Recursos Humanos	SCRUM - Processos/prática
9.1 Desenvolver o Plano de Recursos Humanos	Plano para o tamanho da equipa com base no que é necessário para a duração do projeto ; Plano de sete membros (mais ou menos dois) totalmente dedicados; Dividir o projeto em várias equipas, se o âmbito for grande.
9.2 Constituir a Equipa de Projetos	Desenvolver uma equipa multifuncional dedicado no início do projeto e mantê-la intacta durante a duração do projeto.
9.3 Desenvolver a Equipa de Projeto	Usar valores Scrum (compromisso, transparência, foco, coragem e respeito) para desenvolver e construir equipa; Promover a auto-organização na construção da equipa.
9.4 Gerir a Equipa de Projeto	Facilitar e treinar a equipa Scrum de auto-gestão , fornecendo <i>feedback</i> em tempo real para a equipa Desempenhar o papel de um <i>servant-leader</i>

10 Gestão da Comunicação	SCRUM - Processos/prática
10.1 Identificar os <i>Stakeholders</i>	Identificar as partes interessadas e incorporar um representante comercial (Product Owner) na equipa Scrum
10.2 Planear a Comunicação	Entrega / <i>Sprint backlog</i> e Burndown Charts são indicadores visuais de estado do projeto
10.3 Distribuir Informação	Indicadores visuais de estado do projeto são difusores de informação
10.4 Gerir as Expectativas dos <i>Stackholders</i>	Gestão das partes interessadas é feita por <i>Product Owners</i> que fazem parte da equipa Scrum
10.5 Relatar o Desempenho	Custo e cronograma são estáveis e previsíveis; uso de entrega / <i>Sprint Burndown Charts</i> para mostrar o desempenho em tempo real de desenvolvimento de recurso, ou seja, indicadores visuais de estado do projeto

11 Gestão dos Riscos	SCRUM - Processos/prática
11.1 Planear a Gestão de Risco	Planeamento de risco informal como parte

Anexos

	do plano do <i>Sprint</i> / entrega e reuniões de avaliação. Equipa inteira está envolvida no planeamento de risco, mitigação e resposta.
11.2 Identificar Riscos	Identificar os riscos em <i>Daily Scrum</i> , iteração / plano de entrega e revisão; realizar análise SWOT <i>ad hoc</i> , listas de verificação, Brainstorming
11.3 Análise Qualitativa de Risco	Sem método formal prescrito; matrizes de risco (probabilidade x impacto) podem ser desenvolvidas para riscos especiais, se necessário.
11.4 Análise Quantitativa de Risco	Sem método formal prescrito; matrizes de risco (probabilidade x impacto) podem ser desenvolvidas para riscos especiais, se necessário.
11.5 Planear Respostas a Risco	Alteração (mudança de âmbito ou de recursos), Mitigação (POC), Transferência (<i>outsourcing</i>) e Aceitação (planos de contingência)
11.6 Monitorar os Riscos	Faz parte da equipa de planeamento e revisão

12 Gestão da Aquisições	SCRUM - Processos/prática
12.1 Planear Aquisições	Equipa fornece <i>inputs</i> para a descrição das necessidades de aquisição usando iterações ou prova de conceitos (PICs)
12.2 Conduzir Aquisições	Equipa realiza avaliações e fornece <i>inputs</i> na documentação de contrato
12.3 Gerir Aquisições	Scrum permite contratos com cláusula de rescisão, tipo de contrato conhecido como “ Money for Nothing ”, um cliente pode rescindir o contrato no final de qualquer <i>Sprint</i> , pagando 20-30% do restante valor do contrato “ Change for Free ” - tipo de contrato usado para que um cliente pode fazer alterações no âmbito, sem incorrer em custos adicionais, se alcance total de trabalho contratado não for alterado.
12.4 Fechar Aquisições	Um <i>Sprint</i> adicional (<i>Sprint</i> N +1) pode ser utilizado para o encerramento administrativo formal.